

**ANÁLISIS COMPARATIVO DE TÉCNICAS DE CONTROL CONVENCIONAL E
INTELIGENTE CON LOS MÓDULOS FLEXIBLE LINK, FLEXIBLE JOINT Y
BALL AND BEAM**

**GERMÁN EDUARDO POLANCO ARISTIZÁBAL
OSCAR EDUARDO SOTO CASTAÑEDA**

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA
PROGRAMA INGENIERIA MECATRONICA
SANTIAGO DE CALI
2013**

**ANÁLISIS COMPARATIVO DE TÉCNICAS DE CONTROL CONVENCIONAL E
INTELIGENTE CON LOS MÓDULOS FLEXIBLE LINK, FLEXIBLE JOINT Y
BALL AND BEAM**

**GERMÁN EDUARDO POLANCO ARISTIZÁBAL
OSCAR EDUARDO SOTO CASTAÑEDA**

**Proyecto de Grado para optar el título de
Ingeniero Mecatrónico**

**Director
JESUS ALFONSO LOPEZ SOTELO
Ingeniero Electricista**

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERIA
DEPARTAMENTO DE AUTOMÁTICA Y ELECTRÓNICA
PROGRAMA INGENIERIA MECATRONICA
SANTIAGO DE CALI
2013**

Nota de Aceptación:

Aprobado por el Comité de Grado en cumplimiento de los requisitos exigidos por la Universidad Autónoma de Occidente para optar al título de Ingeniero Mecatrónico.

JIMMY TOMBE

Jurado

JUAN CARLOS MENA

Jurado

Santiago de Cali, 28 de Febrero de 2013

AGRADECIMIENTOS

Al profesor Jesús Alfonso López por su apoyo y aporte en el proceso de realización de la tesis.

CONTENIDO

	pág.
RESUMEN	15
INTRODUCCIÓN	16
1. PROBLEMA DE INVESTIGACIÓN	17
1.1. PLANTEAMIENTO DEL PROBLEMA	17
1.2. FORMULACIÓN DEL PROBLEMA	17
2. JUSTIFICACIÓN	18
3. OBJETIVOS	19
3.1. OBJETIVO GENERAL	19
3.2. OBJETIVOS ESPECÍFICOS	19
4. MARCO DE REFERENCIA	20
4.1. ANTECEDENTES	20
4.2. ANTECEDENTES HISTÓRICOS DE LOS LABORATORIOS VIRTUALES	20
4.3. LABORATORIO VIRTUAL EN LA UNIVERSIDAD AUTÓNOMA DE OCCIDENTE	22
5. MARCO TEORICO	23
5.1. CONTROL CONVENCIONAL	24
5.1.1. Controlador PID	24

5.2. INTELIGENCIA COMPUTACIONAL	25
5.2.1. Control Difuso	26
5.3. LABORATORIOS VIRTUALES	28
5.4. EASY JAVA SIMULATIONS	29
 6. DISEÑO CONTROLADORES	 31
6.1. MODELADO DE LOS SISTEMAS	31
6.1.1. Servo-Planta SRV02	31
6.1.2. Modelado del módulo Flexible Join	32
6.1.3. Modelado del módulo Flexible Link	33
6.1.4. Modelado del módulo Ball and Beam	35
6.2. DISEÑO DE LOS CONTROLADORES DIFUSOS	36
6.2.1. Controlador difuso flexible link	38
6.2.2. Controlador difuso Ball and Beam	41
6.2.3. Controlador difuso Flexible Joint	44
6.3. DISEÑO CONTROLADORES PID	47
6.3.1. Controlador PID Flexible Joint	47
6.3.2. Controlador PID Flexible Link	53
6.3.3. Controlador PID Ball and Beam	59
 7. IMPLEMENTACIÓN CONTROLADORES	 68
7.1. HARDWARE DE QUANSER	68
7.2. ADECUACIÓN PARA TRABAJAR CON LABVIEW	71
7.3. IMPLEMENTACIÓN CONTROLADOR DIFUSO	74
7.3.1. Lectura del ángulo del Flexible Link y Flexible Joint desde labview	74
7.3.2. Implementación de controladores PID	76
7.3.3. Implementación de controladores difusos	79

7.3.4. Implementación del controlador difuso para el módulo Ball and Beam	84
8. VIRTUALIZACION DE LOS MÓDULOS FLEXIBLE LINK Y FLEXIBLE JOINT	87
8.1. VIRTUALIZACIÓN DEL MÓDULO FLEXIBLE LINK	87
8.2. VIRTUALIZACIÓN DEL MÓDULO FLEXIBLE JOINT	90
9. ANÁLISIS COMPARATIVO	93
10. CONCLUSIONES	97
BIBLIOGRAFIA	98
ANEXOS	100

LISTA DE CUADROS

	pág.
Cuadro 1. Reglas controlador difuso Flexible link	40
Cuadro 2. Reglas controlador difuso Ball and Beam	43
Cuadro 3. Reglas controlador difuso Flexible Joint	46
Cuadro 4. Parámetros controladores PID según el método de la oscilación de Ziegler y Nichols	50
Cuadro 5. Valores PID Flexible Joint	53
Cuadro 6. Valores PID Flexible Link	58
Cuadro 7. Valores PID lazo externo Ball and Beam	67
Cuadro 8. Valores PID lazo interno Ball and Beam (Servomotor)	67
Cuadro 9. Especificaciones técnicas de la tarjeta de adquisición de datos NI 6221.	71
Cuadro 10. Índices desempeño Flexible Joint	94
Cuadro 11. Índices desempeño Flexible Link	94
Cuadro 12. Índices desempeño Ball and Beam	95

LISTA DE FIGURAS

	pág.
Figura 1. Controlador de lazo abierto	23
Figura 2. Controlador lazo cerrado	24
Figura 3. Controlador PID	25
Figura 4. Representación de un sistema basado en técnicas de lógica difusa	27
Figura 5. Procesamiento sistema difuso Mamdani	28
Figura 6. Procesamiento sistema difuso Sugeno	28
Figura 7. Simulación desarrollada en Java de un sistema de cuatro tanque	30
Figura 8. Planta servo-motor quanser SRV02	31
Figura 9. Modulo flexible joint	32
Figura 10. Modulo flexible link	34
Figura 11. Sistema bola- viga	35
Figura 12. Bosquejo sistema Ball and Beam	36
Figura 13. Comportamiento del error y su derivada en el Flexible Link	38
Figura 14. Error Flexible Link	39
Figura 15. Derivada del error Flexible Link	39
Figura 16. Salida Controlador Difuso Flexible Link	39
Figura 17. Diagrama bloques controlador difuso Flexible Link	40
Figura 18. Salida del Flexible Link con el controlador difuso	41
Figura 19. Comportamiento del error y su derivada para el Ball and Beam.	41
Figura 20. Error Ball and Beam	42
Figura 21. Derivada del error Ball and Beam	42

Figura 22. Salida controlador difuso Ball and beam	43
Figura 23. Diagrama bloques controlador difuso Ball and Beam	43
Figura 24. Salida del Ball and Beam con el controlador difuso	44
Figura 25. Comportamiento del error y su derivada en el Flexible Joint	44
Figura 26. Error Flexible Joint	45
Figura 27. Derivada del error Flexible Joint	45
Figura 28. Salida Controlador Difuso Flexible Joint	46
Figura 29. Diagrama bloques controlador difuso Flexible Joint	46
Figura 30. Salida del Flexible Joint con el controlador difuso	47
Figura 31. Control PD Flexible Joint	48
Figura 32. Diagrama de bloques PD método analítico Flexible Joint	49
Figura 33. Respuesta método analítico Flexible Joint	49
Figura 34. Método Ziegler y Nichols Flexible Joint	50
Figura 35. Diagrama de bloques PID método ziegler y nichols Flexible Joint	51
Figura 36. Respuesta método ziegler y nichols Flexible Joint	51
Figura 37. Método prueba y error Flexible Joint	52
Figura 38. Diagrama de bloques PID ajuste de ganancias Flexible Joint	52
Figura 39. Respuesta ajuste de ganancias Flexible Joint	53
Figura 40. Control PD Flexible Link	54
Figura 41. Diagrama de bloques PD método analítico Flexible Link	55
Figura 42. Respuesta método analítico Flexible Link	55
Figura 43. Método Ziegler y Nichols Flexible Link	56
Figura 44. Diagrama de bloques PID método ziegler y nichols Flexible Link	56
Figura 45. Respuesta método ziegler y nichols Flexible Link	57

Figura 46. Método prueba y error Flexible Link	57
Figura 47. Diagrama de bloques PID ajuste de ganancias Flexible Link	58
Figura 48. Respuesta ajuste de ganancias Flexible Link	58
Figura 49. Controlador PD Ball and Beam	59
Figura 50. Diagrama de bloques PD método analítico lazo externo Ball and Beam	60
Figura 51. Respuesta método analítico lazo externo Ball and Beam	60
Figura 52. Controlador PD lazo interno Ball and Beam (Servomotor)	61
Figura 53. Diagrama de bloques PD método analítico lazo interno Ball and Beam (Servomotor)	62
Figura 54. Respuesta método analítico lazo interno Ball and Beam (Servomotor)	62
Figura 55. Método Ziegler y Nichols lazo externo Ball and Beam	63
Figura 56. Método Ziegler y Nichols lazo interno Ball and Beam (Servomotor)	63
Figura 57. Diagrama de bloques PID método ziegler y nichols lazo externo Ball and Beam	64
Figura 58. Respuesta método ziegler y nichols lazo externo Ball and Beam	64
Figura 59. Diagrama de bloques PID método ziegler y nichols lazo interno Ball and Beam (Servomotor)	65
Figura 60. Respuesta método ziegler y nichols lazo interno Ball and Beam (Servomotor)	65
Figura 61. Diagrama de bloques PID método prueba y error lazo externo Ball and Beam	66
Figura 62. Respuesta método prueba y error lazo externo Ball and Beam	66
Figura 63. Diagrama de bloques PID ajuste ganancias lazo interno Ball and Beam (Servomotor)	66
Figura 64. Respuesta ajuste ganancias lazo interno Ball and Beam (Servomotor)	67

Figura 65. UPM-15-03	68
Figura 66. Conectores tipo 6 pin mini DIN a 6 pin mini DIN	69
Figura 67. Quanser Q8 terminal board	69
Figura 68. Conector tipo 5-pin-DIN to 4xRCA	70
Figura 69. Conector 5-pin-stereo-DIN to 5-pin-stereo-DIN	70
Figura 70. Esquema de conexión de las señales de los módulos	71
Figura 71. Adquisición señales Tarjeta NI 6221	73
Figura 72. Configuración de la conexión para adquisición de las señales de encoders en NI 6221	73
Figura 73. Sumatoria de los ángulos θ y α para el modulo flexible Link	75
Figura 74. Sumatoria de los ángulos θ y α para el modulo flexible Joint	75
Figura 75. Implementación acción proporcional	76
Figura 76. Implementación acción derivativa	77
Figura 77. Implementación acción integral	77
Figura 78. Respuesta y acción de control controlador PID Flexible Joint	78
Figura 79. Respuesta y acción de control controlador PID Flexible Link	78
Figura 80. Respuesta y acción de control controlador PID Ball and Beam	79
Figura 81. Campanas gaussianas error Flexible Link y Flexible Joint	80
Figura 82. Campanas gaussianas derivada del error	80
Figura 83. Nodo de fórmula que representa los conjuntos difusos del error	82
Figura 84. Nodo de fórmula que representa los conjuntos difusos de la derivada del error	82
Figura 85. Respuesta y acción de control controlador difuso Flexible Link	83
Figura 86. Respuesta y acción de control controlador difuso Flexible Joint	83
Figura 87. Campanas gaussianas error Ball and Beam	84

Figura 88. Nodo de fórmula que representa los conjuntos difusos del error en el Ball and Beam	85
Figura 89. Nodo de fórmula que representa los conjuntos difusos de la derivada del error en el ball and beam	85
Figura 90. Respuesta y acción de control controlador difuso Ball and Beam	86
Figura 91. Ventana principal Flexible Link	88
Figura 92. Ventana secundaria Flexible Link.	88
Figura 93. Panel Control	89
Figura 94. Referencia	89
Figura 95. Animación Flexible Link	90
Figura 96. Gráficas.	90
Figura 97. Ventana principal Flexible Joint	91
Figura 98. Ventana secundaria Flexible Joint	91
Figura 99. Animación del Flexible Joint.	92
Figura 100. Gráfica Índices desempeños Flexible Joint	94
Figura 101. Gráficas Índices de desempeño Flexible Link	95
Figura 102. Gráficas Índice de desempeño Ball and Beam	95
Figura 103. Tercer orden Flexible Joint y respuesta ante una entrada.	100
Figura 104. Tercer orden Flexible Link y respuesta ante una entrada.	101

LISTA ANEXOS

	pág.
Anexo A. Flexible Joint Segundo Orden	100
Anexo B. Flexible Link segundo orden	101
Anexo C. Código para generar cada uno de los conjuntos difusos Flexible Link y Flexible Joint	102
Anexo D. Código para generar cada uno de los conjuntos difusos Ball and Beam	103

RESUMEN

En este trabajo se presenta el diseño de dos tipos de controladores un PID y un difuso para cada uno de los módulos Flexible Link, Flexible Joint y Ball and beam. Además se implementan los controladores en programa Labview para cada uno de estos, en esta parte se observa la independización que se hizo de la tarjeta de Quanser a la tarjeta de National Instrument. También se realiza una virtualización del flexible joint y flexible link en el programa Easy Java, por último se realiza un análisis comparativo de los controladores diseñados e implementados con los índices de desempeño.

Palabras claves (Ball and Beam, Control difuso, Control PID, índices desempeño, flexible joint, flexible link, virtualización)

INTRODUCCIÓN

Las técnicas de control convencional e inteligente tienen como objetivo dirigir o regular el comportamiento de un sistema o proceso a partir de la modificación de algunas de las entradas. Entre las técnicas de control convencional se puede mencionar el PID y sus variaciones (controlador proporcional, controlador derivativo, controlador integral, controlador PID, etc.) y entre las técnicas del control inteligente se destacan el control neuronal, y el control difuso.

Entre las características más recurrentes en el control convencional se encuentra que se utiliza un modelo del sistema a controlar dado generalmente en forma de ecuaciones diferenciales o de una función de transferencia mientras, en general las técnicas de control inteligente no necesitan un modelo de la planta pues lo que necesitan son datos del comportamiento de la entrada y de la salida del sistema o el conocimiento lingüístico del funcionamiento del sistema a controlar.

En los últimos años las denominadas TICs se han vuelto una herramienta con bastante proyección en la enseñanza y un área como el control automático no ha sido ajeno a esta situación. Con este trabajo se pretende complementar con los módulos flexible link y flexible joint, un proyecto realizado anteriormente donde se trabajaron los procesos de la planta de nivel, el servomotor y el ball and beam. En este proyecto se realizó la virtualización de los módulos mencionados además de la implementación de los controladores convencionales (PID) e inteligente (difusos) en los módulos reales para compararlos por medio de índices de desempeño.

1. PROBLEMA DE INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA

El incremento de las demandas tecnológicas en nuestros tiempos, han generado sistemas complejos que requieren controladores altamente sofisticados para asegurar alto desempeño dentro de las condiciones adversas. Estas y otras condiciones de control no se pueden cumplir con los controladores convencionales, debido principalmente a la falta de conocimiento preciso acerca del proceso a controlar. La adquisición de conocimiento adecuado del sistema en ocasiones es problemática debido a la complejidad del sistema y al hecho de la estructura y los parámetros en muchos sistemas cambian de manera significativa e impredecible con el tiempo. Es bajo estas condiciones en donde se utilizan las técnicas de control inteligente [2].

La Universidad Autónoma de Occidente adquirió los módulos de flexible joint, flexible link y ball and beam para el apoyo en los cursos de pregrado y postgrado del área de automática para su desarrollo académico en métodos de control actual. El desarrollo de aplicaciones para estos módulos de control actualmente se realiza bajo la interfaz dotada por Quanser, la universidad requiere que las aplicaciones sean desarrolladas bajo interfaces más comerciales o conocidas y de fácil uso por los estudiantes.

En base a que estos módulos son sistemas no lineales y que se les puede realizar diferentes tipos de control, se plantea la siguiente pregunta:

1.2. FORMULACIÓN DEL PROBLEMA

¿Los controladores inteligentes tendrán mejor funcionamiento que los controladores convencionales para estos módulos?

2. JUSTIFICACIÓN

Con la evolución de la tecnología en los campos de la comunicación, información y el desarrollo de mecanismos cada vez más eficientes y complejos, el enfoque de la educación ha tratado de expandirse buscando la incursión de nuevas formas de hacer llegar el conocimiento a las personas, una de ellas es la creación de entornos virtuales que permitan emular los comportamientos y situaciones que en ocasiones no se pueden presentar en forma tangible o física en cuanto a desarrollo de experimentos de laboratorio se refiere.

Actualmente la Universidad Autónoma de Occidente en el programa de Ingeniería Mecatrónica ha incluido dentro de su ajuste curricular el curso de control inteligente lo que hace también que este proyecto implante bases significativas para el desarrollo de esta asignatura.

El presente proyecto se enfoca en el estudio y análisis de las diferentes técnicas de control que existen en la actualidad ya que estas juegan un papel muy importante a la hora de optimizar procesos y aprovechar al máximo las capacidades que pueda tener una planta, para ello se pretende desarrollar en el ámbito educativo un análisis del control clásico y el control inteligente que nos permita visualizar de una manera mucho más clara cuál de ellas obtiene un mejor desempeño en el control de los módulos flexible link, flexible joint y ball and beam desarrollados por Quanser.

Además se tendrá en cuenta que en proyectos pasados se ha logrado ilustrar por medio de laboratorios virtuales el funcionamiento de plantas presentes en nuestros laboratorios. La creación de un laboratorio virtual de control donde se ilustre al usuario los comportamientos de las plantas seleccionadas después de implementar las técnicas de control ayuda a complementar los trabajos realizados con anterioridad por otros estudiantes de ingeniería mecatrónica.

3. OBJETIVOS

3.1. OBJETIVO GENERAL

Implementar y comparar algunas técnicas de control clásico e inteligente a implementar en los módulos flexible link, flexible joint y ball and beam.

3.2. OBJETIVOS ESPECÍFICOS

Estudiar las técnicas de control inteligente.

Seleccionar las técnicas de control a implementar en los módulos flexible link, flexible joint y ball and beam.

Diseñar un módulo virtual para la planta Quanser flexible Joint y Quanser Flexible Link.

Implementar las técnicas de control convencional e inteligente en los módulos flexible link, flexible joint y ball and beam.

Realizar la comparación entre las técnicas de control implementadas con índices de desempeño que midan el comportamiento de la acción de control y del error de seguimiento.

4. MARCO DE REFERENCIA

4.1. ANTECEDENTES

El control inteligente es una generalización del concepto de control y se puede ver como un campo dentro de la disciplina de control. El control inteligente es la disciplina donde los métodos de control se desarrollan para emular las características importantes del ser humano. Estas características incluyen adaptación y aprendizaje, planeación bajo gran incertidumbre y el trabajo con gran cantidad de datos.

Un sistema de control inteligente es autónomo esto significa que tiene el poder de autogobernarse. Existen varios grados de autonomía: un controlador totalmente autónomo debería tener la habilidad de aún reparar su propio hardware si uno de sus componentes falla. Un control fijo convencional se considera de bajo grado de autonomía: un control adaptivo convencional tiene un alto grado de autonomía. La autonomía es el objetivo en los sistemas de control complejos y los controladores inteligentes son una manera de lograrlo [2].

4.2. ANTECEDENTES HISTÓRICOS DE LOS LABORATORIOS VIRTUALES

En el año 1984 aparece el concepto de instrumento virtual como instrumento cuyas características vienen definidas por la programación, este concepto y su aplicación en distintas áreas de laboratorio son "*el primer paso hacia los laboratorios virtuales*". Durante los años posteriores se fueron lanzando distintas propuestas para laboratorios, entre ellas la de un laboratorio de control de sistemas en 1991 en la Universidad de Bucknell en E.E.U.U., que se fue desarrollando a lo largo de los años hasta convertirse, años más tarde, en un sistema de procesamiento digital de señal y conexión a Internet. Sin embargo, no es hasta el año de 1992 donde aparece explícitamente el término de laboratorio virtual (en este caso describiendo la programación orientada a objetos para el desarrollo de un laboratorio de simulación) con un prototipo denominado MWS (Microscopist's Workstation) que ha evolucionado al actual sistema conocido como CMDA (Collaboratory for Microscopio Digital Anatomy).

A lo largo de 1993, lo más interesante a destacar es la aparición de un nuevo concepto, íntimamente ligado al de laboratorio virtual, que es el *colaboratorio*, definido como la combinación de intereses de la comunidad científica en general

con los ingenieros o informáticos para crear sistemas de comunicación y cálculo con herramientas que soporten la colaboración científica. Ya en 1994 se presenta un estudio realizado por la Universidad de Vanderbilt en E.E.U.U. en el que se desarrolla un laboratorio virtual basado en simulación como apoyo a las prácticas tradicionales y que concluyó con la necesidad de esta herramienta para aprender las habilidades básicas y el manejo de los equipos, lo cual optimizaba tanto el tiempo de los alumnos como el del personal de laboratorio.

Pasando a 1995, encontramos una descripción detallada de los requisitos que un laboratorio virtual debe cumplir encontrando, a la vez, una serie de similitudes con las características de otra tecnología Web desarrollada en los últimos años. En la conferencia IMTC (IEEE Instrumentation and Measurement Technology Conference) celebrada en 1996 empiezan a presentarse distintos aspectos de lo que es un laboratorio virtual, y aparecen varios artículos donde se indica los elementos necesarios para el éxito de un sistema de educación a distancia y se definen las características que debía tener un laboratorio virtual.

En 1997, en la conferencia del IMTC se hace un repaso de normas relativas a instrumentos virtuales, además de algunos ejemplos de control a través de Internet empleando Visual Basic o Java, una descripción de pros y contras de la simulación aplicada a laboratorios y el desarrollo de un bus virtual para instrumentación que permite interconectar a través de redes telemáticas distintos dispositivos GPIB. Este mismo año, investigadores de la Universidad de Illinois presentan un completo laboratorio de instrumentación electrónica puesto a disposición de los usuarios a través de Internet. Este es el primer laboratorio virtual con control remoto de instrumentación electrónica en funcionamiento.

A lo largo de 1998, las referencias a laboratorios virtuales incrementan su número de forma considerable, entre las más destacadas encontramos la conferencia de IMTC y la asociada ETIMVIS'98 donde se describe un modelo detallado de un laboratorio virtual de simulación y una propuesta de laboratorio docente en el que los alumnos utilizan instrumentos virtuales para crear sus experimentos. En 1999, en el IMTC se hace una detallada especificación de cómo montar un laboratorio virtual con elementos comerciales disponibles, junto con los requisitos básicos que debe afrontar cualquiera que se plantee el diseño de un laboratorio virtual.

A partir del año 2000 se van sucediendo artículos en conferencias y revistas donde se describen distintos laboratorios virtuales y donde se va entrando en detalle en los distintos métodos a utilizar en el desarrollo de laboratorios virtuales y se comentan posibles soluciones que mejoren o aumenten el rendimiento de estos. La idea de utilizar la simulación como paso previo al uso de los instrumentos permite que se reduzca el tiempo necesario de uso del instrumento

real y por tanto, del recurso más costoso. Durante años se han ido experimentando soluciones de comunicación por Internet empleando distintas tecnologías como CGI, ISAPI, COM+, o los últimos desarrollos para intercomunicación multimedia con ConferenceXP.

4.3. LABORATORIO VIRTUAL EN LA UNIVERSIDAD AUTÓNOMA DE OCCIDENTE

En la Universidad se desarrolló un trabajo sobre un Laboratorio Virtual de Control Inteligente, que tuvo como objetivo general el diseño e implementación del mismo como apoyo al sistema de educación superior [4]. Otros objetivos de este laboratorio virtual fueron:

- Estudiar las diversas técnicas de control inteligente que se van a implementar.
- Definir las plantas a las que se va implementar el control inteligente.
- Diseñar los controladores basados en redes neuronales y sistemas de inferencia difusos haciendo uso del software MATLAB.
- Diseñar la aplicación (laboratorio virtual).
- Elaborar el algoritmo y las interfaces gráficas que permita simular el control inteligente en el entorno de programación escogido.
- Simular y probar los sistemas de control inteligente.

En este proyecto se seleccionaron las plantas:

- Planta de nivel
- Servomotor
- Bola Viva

A cada una de estas plantas se le diseñó un control PID, un control difuso y control por Redes Neuronales. También se diseñaron las interfaces gráficas para el laboratorio virtual donde cada una contenía una animación de la planta, los controladores, las señales temporales y la referencia.

También se desarrolló un trabajo sobre Laboratorio Virtual de Redes Neuronales [8] y el desarrollo de una Librería de Control Difuso para los PLC de Siemens [9]

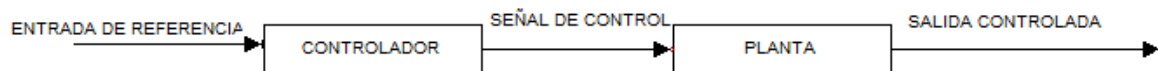
5. MARCO TEORICO

Un sistema de control está compuesto por una planta la cual es el proceso o sistema al cual se le pretende actuar para establecer el control. El controlador que gestiona el valor de la salida de control a partir de una entrada de referencia, la cual es una señal externa aplicada al controlador como un parámetro de ajuste. La señal de control que es una variable manipulada por el sistema de control aplicada a la planta y la salida controlada que es la variable de salida de la planta que está siendo controlada.

Los sistemas de control se clasifican según su estructura: lazo abierto o lazo cerrado.

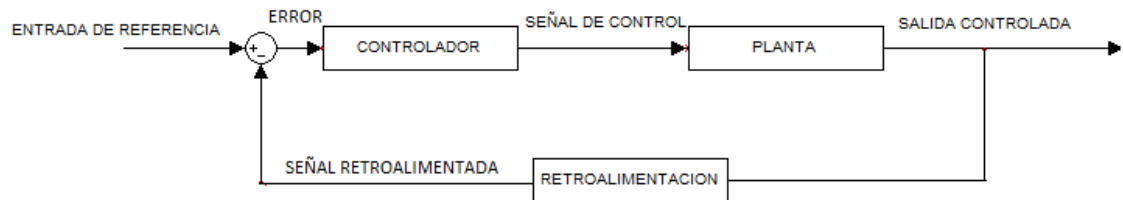
En los sistemas de control de lazo abierto el controlador es independiente de la salida controlada de la planta, están compuestos únicamente por el controlador, la planta, la entrada de referencia al controlador, la señal de control de la planta y la salida controlada, como se observa en la Figura 1. En estos sistemas la referencia debe tener una calibración muy precisa ya que no se puede hacer una corrección dinámica de la señal controlada de forma automática. [7]

Figura 1. Controlador de lazo abierto



En los sistemas de control de lazo cerrado se aplica la retroalimentación para ofrecer mayor capacidad de actuación a los controladores, esta permite que la salida controlada se compare con la entrada de referencia, de forma que el controlador actúe de forma apropiada considerando la diferencia producida entre ambos (error). Este sistema está compuesto por la entrada de referencia, la señal del error, el comparador, el controlador, la señal de control, la planta, la salida controlada, la retroalimentación y la señal de retroalimentación, como se observa en la Figura 2. [7]

Figura 2. Controlador lazo cerrado



5.1. CONTROL CONVENCIONAL

En los sistemas de control convencional existen varios tipos de controladores según su funcionamiento y estructura, entre los cuales se encuentran:

- Servosistemas reguladores: Permiten que la salida controlada se mantenga constante frente a cambios exteriores.
- Servosistemas de posición: Permiten que la salida controlada siga de forma continua los valores establecidos por la entrada de referencia.
- Control modelo-referencia: Son esquemas de control no lineal donde a partir de la entrada se utiliza un modelo matemático para simular el comportamiento de la planta. La entrada se compara con la salida de la planta, la diferencia de estas genera la señal de control.[7]
- Control adaptativo: Tiene en cuenta un conjunto de características dinámicas de la planta que afecta a unos parámetros configurables del controlador. Las variables de la planta son analizadas, mediante un proceso de decisión, se establecen las modificaciones de los parámetros del módulo controlador con el fin de obtener un funcionamiento óptimo. [7]

5.1.1. Controlador PID. Es un controlador de lazo cerrado, donde la señal de control depende de la señal de error en función de su comportamiento y variación de la señal, esto permite realizar un control más estable y preciso.

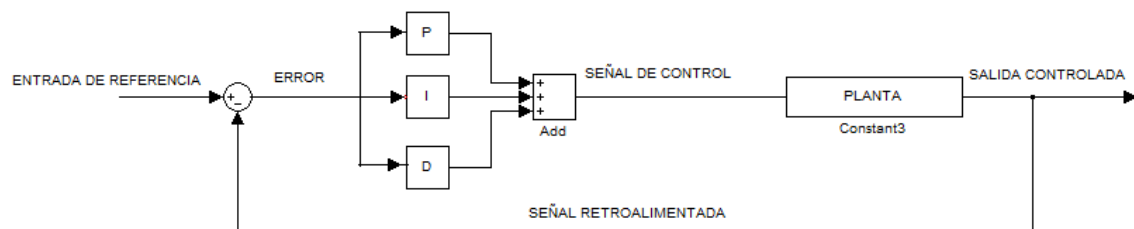
La señal de control se calcula de por medio de tres medidas para procesar la señal de salida del controlada, anular el error del instante actual, medir la cantidad

de tiempo que la señal se ha mantenido con valores incorrectos e intentar anticipar errores futuros midiendo la tasa de variación de la señal en cada instante.

- **Módulo Proporcional:** se usa para tener en cuenta el error actual, la señal de error se multiplica por una constante para ampliar su valor y así obtener una componente de la señal de error que actúe de forma proporcional al error.[7]
- **Módulo Integral:** se usa para tener en cuenta los valores de error ocurridos en el pasado, se realiza una operación de integración en el intervalo de tiempo determinado, es decir se calcula la tasa de los valores de error medidos en el pasado. Este valor se suma al valor del módulo proporcional, evitando grandes oscilaciones debido a picos en breves periodos de tiempo de la señal de error.[7]
- **Módulo Derivativo:** se usa para prevenir errores futuros, este módulo calcula la primera derivada de la señal de error. Este valor se considera como una medida de la respuesta del sistema a los cambios. Así cuanto mayor es la derivada más rápido responde el controlador a los cambios del sistema. [7]

En la Figura 3 se observa el modelo de un controlador PID.

Figura 3. Controlador PID



5.2. INTELIGENCIA COMPUTACIONAL

La inteligencia computacional surge de la necesidad de imitar los procesos de la inteligencia a un alto nivel (razonamiento, deducción y lógica). Es por ello que surgen unas nuevas ramas del control como lo son el control por redes neuronales

artificiales (RNA), la lógica difusa (LD), y el control por algoritmos genéticos (AG), todos estos desarrollados con técnicas de naturaleza bioinspirada. Se le llama inteligencia computacional porque básicamente el procesamiento final de la información se hace de manera numérica (Ideales para simularse en sistemas de cómputo).

El objetivo de un sistema de control automático es encontrar un dispositivo externo al sistemas dinámico a controlar (el controlador o compensador) que proporcione una señal adecuada al mismo para que la salida del sistema tenga características deseadas.

El Control Inteligente surge cuando para diseñar o implementar el controlador se utiliza alguna técnica bioinspirada normalmente del ámbito de la Inteligencia Computacional (RNA, LD, AG). Algunas de las características de este tipo de control son: autonomía, operación continua, razonamiento con incertidumbre y datos incompletos, eficiencia computacional, entre otras

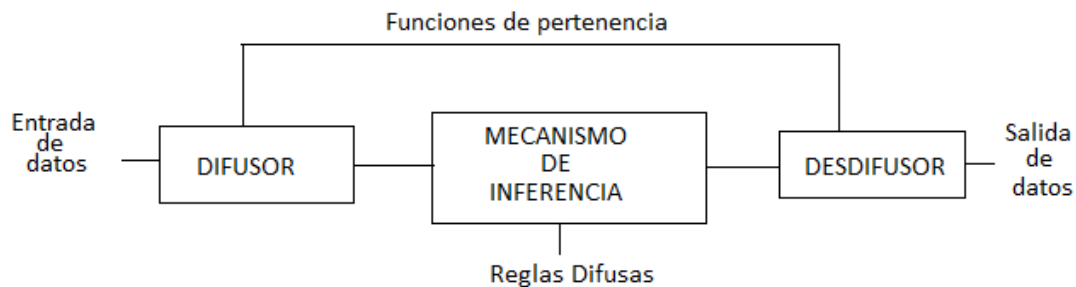
En la actualidad las metodologías del control inteligente están siendo aplicadas a la robótica, las comunicaciones, la manufactura, el control de tráfico, por mencionar algunas pocas. Las áreas donde se está realizando trabajo alrededor de este control son: redes neuronales, control difuso, algoritmos genéticos, sistemas de planeación, sistemas expertos y sistemas híbridos [14].

Para el desarrollo de este proyecto se debe tener en cuenta especialmente el control con redes neuronales y el control difuso, ya que estas son las estrategias de control a implementar en las plantas escogidas para desarrollar la representación virtual y el control de las plantas flexible link y flexible joint.

5.2.1. Control Difuso. La lógica difusa es una metodología que proporciona una manera simple y elegante de obtener una conclusión a partir de información de entrada vaga, ambigua, imprecisa, con ruido o incompleta, en general la lógica difusa imita como una persona toma decisiones basada en información con las características mencionadas. Una de las ventajas de la lógica difusa es la posibilidad de implementar sistemas basados en ella tanto en hardware como en software o en combinación de ambos. El control difuso o control por lógica difusa tuvo su origen en 1965 cuando Zadeh planteó la teoría de conjuntos difusos. Al igual que cualquier otro sistema de control, el control por lógica difusa trabaja sobre un estado deseado o setpoint, establece el valor requerido del sistema por medio de una variable medida y lleva a cabo su tarea sobre una variable controlada como se muestra en la Figura 4.

La diferencia entre un sistema de control convencional y uno que utiliza lógica difusa se encuentra en la forma de realizar el proceso de inferencia del valor de salida, en un controlador difuso esta respuesta se calcula por medio de un algoritmo difuso en el cual se encuentran comprendidas una serie de etapas encargadas de determinar la respuesta adecuada. Los sistemas de inferencia difusa se basan en el lenguaje natural o cotidiano, ésta condición les permite realizar una clasificación de la información más que una cuantificación; el proceso de inferencia de este tipo de sistemas vincula en un alto grado el sentido común y el conocimiento del experto codificados en forma de reglas heurísticas que determinarán las acciones de control. [13]

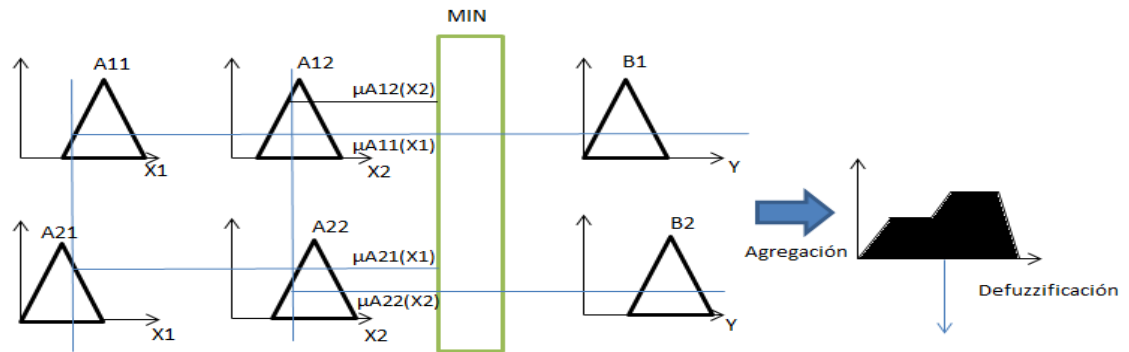
Figura 4. Representación de un sistema basado en técnicas de lógica difusa



En la Figura 4 se aprecian tres bloques, uno etiquetado como difusor, encargado de transformar las variables numéricas a conjuntos difusos, otro etiquetado como desdifusor cuyo objetivo es convertir las variables difusas a numéricas. El bloque de inferencia viene acompañado de una base reglas difusas, por lo que el diseño de un controlador difuso involucra la construcción de reglas de control, en muchos casos, se pueden obtener estas reglas describiendo las acciones de los operadores del sistema de control. Una vez establecidas la base de reglas se puede realizar la estrategia de control o mecanismo de inferencia usando razonamiento difuso. Existen diferentes tipos de sistemas de inferencia difusa entre los que se destacan el sistema de inferencia tipo Mamdani y Sugeno [4].

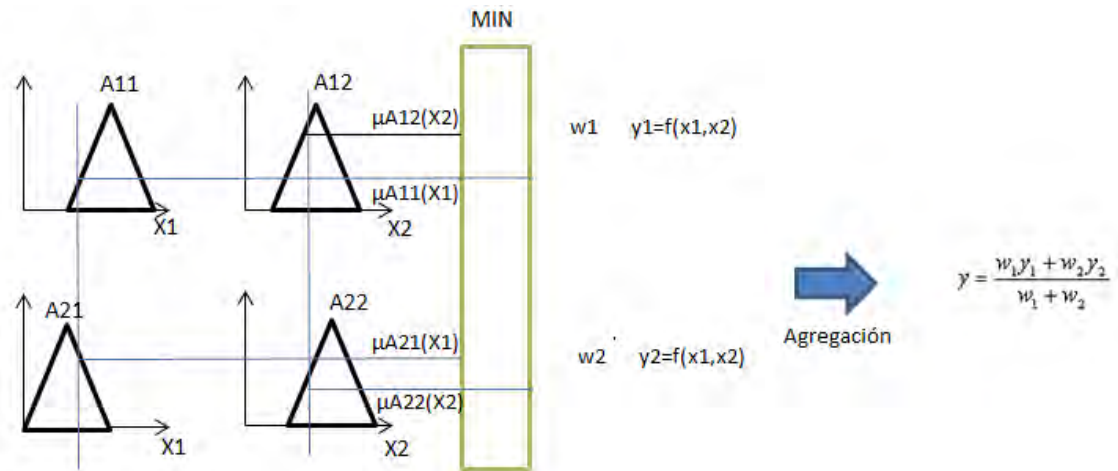
Sistema de inferencia tipo Mamdani: En este tipo de sistema tanto las entradas como las salidas se pueden definir como conjuntos difusos, lo que trae como ventaja el utilizar completamente la información lingüística para describir el problema a solucionar. Cabe resaltar, que en este sistema existen varios métodos para realizar la defuzzificación siendo el más utilizado la técnica del centro de masa (COG) [4].

Figura 5. Procesamiento sistema difuso Mamdani



Sistema de inferencia tipo Sugeno: En este sistema de inferencia las entradas se definen como conjuntos difusos, pero las salidas se definen en función de las entradas, usualmente como una combinación lineal. La principal ventaja de este sistema es su simplicidad a la hora de ser implementado, ya que no se hace necesario llevar a cabo el proceso de defuzzificación ya que la salida al estar definida por funciones es un valor numérico [4].

Figura 6. Procesamiento sistema difuso Sugeno



5.3. LABORATORIOS VIRTUALES

En un laboratorio virtual, los computadores son utilizados para simular el comportamiento de los sistemas en estudio mediante modelos matemáticos. A pesar de que en este tipo de laboratorios no se interactúa con la planta real, la

simulación con modelos simulados es comparable siempre y cuando se cumpla con las siguientes premisas:

Sean utilizados modelos matemáticos realistas que representen al alumno los detalles importantes del sistema a analizar

Se complementen las gráficas que muestran la evolución temporal de los sistemas con animaciones que permitan al alumno visualizar y entender el comportamiento del sistema

El uso de laboratorios virtuales tiene algunas ventajas importantes con respecto a los laboratorios reales (presenciales o remotos). Dado que un laboratorio virtual se basa en un modelo matemático que es ejecutado en un computador, su configuración y puesta a punto es mucho más sencilla con respecto a los laboratorios reales. Además presentan un grado de robustez y seguridad mucho más elevado ya que al no haber dispositivos reales estos no pueden causar problemas en el entorno. Sin embargo como inconveniente con respecto a los laboratorios reales cabe señalar que los laboratorios virtuales se encuentran limitados por el modelo y para poder ser manejable tienden a simplificarse, con lo cual se pierde información con respecto al sistemas real. Una forma de solucionar este problema, es utilizar los Laboratorios Virtuales junto con los Laboratorios

Reales de tal forma que los alumnos primero relazarían las prácticas sobre los Laboratorios virtuales para pasar posteriormente al Laboratorio Real (presencial o remoto) [1]

5.4. EASY JAVA SIMULATIONS

Es una herramienta diseñada para la creación de simulaciones por ordenador discreto. Una simulación por ordenador discreto, o simplemente una simulación por ordenador, es un programa informático que intenta reproducir, con fines didácticos o científicos, un fenómeno natural a través de la visualización de los diferentes estados que puede tener. Cada uno de estos estados se describe mediante un conjunto de variables que cambian con el tiempo debido a la iteración de un algoritmo dado.

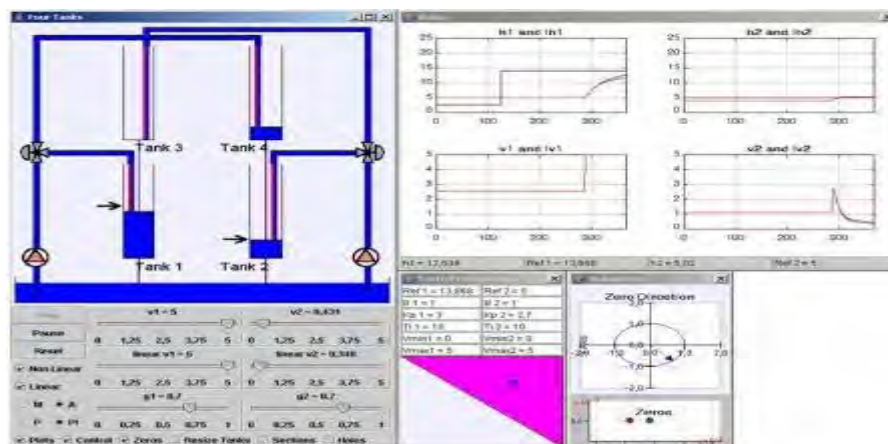
Todo esto significa que EJS es un programa que le ayuda a crear otros programas, más precisamente, para crear simulaciones científicas. Debido a que

existe un valor educativo en el proceso de creación de una simulación, EJS también puede ser utilizado como una herramienta pedagógica en sí [6].

Ejemplo aplicación laboratorios virtuales

Laboratorios virtuales del tipo monolítico o híbrido desarrollados por el departamento de informática y automática de la UNED. La Figura 7 representa la simulación de un sistema de control de cuatro tanques interconectados desarrollada íntegramente en Java con el entorno Easy Java Simulations, es decir la interfaz y el modelo constituyen un mismo programa, en este caso, un applet de Java. Aquí, la interactividad se traduce en la posibilidad de modificar, antes o durante un experimento, los niveles del líquido, el porcentaje de apertura de las válvulas, la geometría de los tanques, el diámetro de los desagües, los parámetros de los dos controladores, etc. [5]

Figura 7. Simulación desarrollada en Java de un sistema de cuatro tanques



Fuente: Ejemplo laboratorios virtuales [en línea] Laboratorio Virtual. España. [Citado el 19 de noviembre de 2012] <http://lab.dia.uned.es/rlab/contenido/labvirtual.html?page=3>

6. DISEÑO CONTROLADORES

6.1. MODELADO DE LOS SISTEMAS

Para realizar el diseño de las técnicas de control convencional e inteligente para los módulos flexible link, flexible joint y ball and beam es necesario considerar conceptos y modelos matemáticos de estos módulos.

6.1.1. Servo-Planta SRV02. El SRV02 hace parte fundamental del resto de los módulos, por medio del servomotor al acoplarse en el flexible link, flexible joint y ball and beam, estos pueden moverse y realizar el control que se desee.

El servomotor de CC está montado en un bastidor de aluminio sólido como se muestra en la figura 8. El motor de fabricación suiza incorpora una caja de cambios de 14:01, cuya salida lleva a un equipo externo. El motor acciona un engranaje conectado a un eje de salida independiente que gira en un cojinete de aluminio mecanizado con alta precisión. El eje de salida está equipado con un codificador. Esta segunda velocidad en el eje de salida acciona un engranaje anti desajuste conectado a un potenciómetro de precisión. El potenciómetro se utiliza para medir el ángulo de salida. La relación de engranaje externo se puede cambiar 01:01-05:01 utilizando distintos engranajes [12].

Figura 8. Planta servo-motor quanser SRV02



Fuente: Servo-Planta SRV02 [en línea] Rotary Control Challenge. Estados Unidos. [Citado el 19 de noviembre de 2012]
http://www.quanser.com/english/downloads/products/Rotary/SRV02_PIS_030608.pdf

6.1.2. Modelado del módulo Flexible Joint. Este módulo de rotación diseñado por Quanser consta de un brazo libre unido a dos muelles idénticos como se muestra en la figura 9. Los muelles están montados en un chasis de aluminio que es sujeto al engranaje de carga SRV02. La articulación flexible permite realizar experimentos que describen y modelan el comportamiento de una articulación flexible en un robot o una nave espacial. Este experimento también es útil en el estudio de análisis de vibraciones y resonancias. El sistema de articulación flexible es similar en la naturaleza de los problemas de control a las grandes articulaciones de los robots orientados donde la flexibilidad se exhibe en la caja de cambios [10]

Figura 9. Modulo flexible joint



Fuente: Flexible joint Quanser. [en línea] Flexible Joint. Estados Unidos. [Consultado el 19 de noviembre de 2012] http://www.quanser.com/english/html/products/fs_product_challenge.asp?lang_code=english&pcat_code=exp-rot&prod_code=R4-flexjnt

El modelo espacio estado para esta planta se representa de la siguiente forma:

$$\dot{x} = Ax + Bu \quad (1)$$

$$y = Cx \quad (2)$$

Dónde:

$$u = Vin \quad (3)$$

$$y = [\theta \quad \alpha \quad \dot{\theta} \quad \dot{\alpha}]^T \quad (4)$$

La salida es $y = \theta + \alpha$, donde θ es el ángulo de salida DC del eje del servo y α es el ángulo relativo del brazo al ángulo servo DC eje de salida, es decir es la medición de la deflexión angular del brazo y Vin es el voltaje que arroja el motor.

El modelo matemático por medio de variables de estado que define el comportamiento del módulo flexible joint se encuentra representado en las siguientes matrices (Ecuaciones 5, 7 y 8):

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 767.05 & -52.8 & 0 \\ 0 & -1040.1 & 52.8 & 0 \end{bmatrix} \quad (5)$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 98.33 \\ -98.33 \end{bmatrix} \quad (6)$$

$$C = [1 \quad 1 \quad 0 \quad 0] \quad (7)$$

Por medio de la transformación de variables de estado a función de transferencia se obtiene, la función de transferencia que representa el comportamiento de la planta, teniendo como resultado (Ecuación 8):

$$\frac{\theta(s)}{V(s)} = \frac{26850}{s^4 + 52.8s^3 + 1040s^2 + 14410s} \quad (8)$$

6.1.3. Modelado del módulo Flexible Link. Este módulo de rotación de quanser está diseñado para ser acoplado a la planta SRV02 y consta de una fina regla de acero inoxidable equipada con un sensor de deformación como se muestra en la figura 10. La deflexión del brazo se mide a través de la salida del sensor de deformación. El modelo está diseñado para acentuar los efectos de enlaces flexibles en los sistemas de control de un robot. Tal flexibilidad es de naturaleza similar a los problemas de control que se encuentran en grandes estructuras espaciales donde las limitaciones de peso dan como resultado estructuras flexibles que deben ser controlados usando técnicas de realimentación. Un motor de corriente continua gira un enlace flexible desde un extremo en el plano horizontal. El extremo del motor del enlace está instrumentado con un sensor de deformación que puede detectar la deflexión de la punta [11]

Figura 10. Modulo flexible link



Fuente: Flexible link Quanser. [en linea] Flexible Link. Estados Unidos. [Consultado el 19 de noviembre de 2012]
http://www.quanser.com/english/html/products/fs_product_challenge.asp?lang_code=english&pcat_code=exp-rot&prod_code=R3-flexlink

Este módulo es similar al joint, lo único que cambia es que el link es más flexible que el joint. El modelo matemático por medio de variables de estado que define el comportamiento del módulo se encuentra representado en las siguientes matrices (Ecuaciones 9, 10 y 11):

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 2165 & -53.6 & 0 \\ 0 & -2797 & 53.6 & 0 \end{bmatrix} \quad (9)$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 99 \\ -99 \end{bmatrix} \quad (10)$$

$$C = [1 \quad 1 \quad 0 \quad 0] \quad (11)$$

Por medio de la transformación de variables de estado a función de transferencia se obtiene la función de transferencia que representa el comportamiento de la planta, teniendo como resultado (Ecuación 12):

$$\frac{\theta(s)}{V(s)} = \frac{62570}{s^4 + 53.6s^3 + 2797s^2 + 33880s} \quad (12)$$

6.1.4. Modelado del módulo Ball and Beam. El sistema de control Bola-Viga (figura 11) consiste en una viga sujeta a un servomotor (SRV-02) que controla su movimiento y sobre la cual corre una bola de acero inoxidable. Adicionalmente utiliza otra viga frente al motor cuya función es sensar la posición, enviando distintos voltajes dependiendo de dónde se encuentre la bola de referencia. El objetivo es controlar que la bola en la viga superior siga en todo momento la posición de la bola de referencia que corre en la viga inferior.

Figura 11. Sistema bola- viga

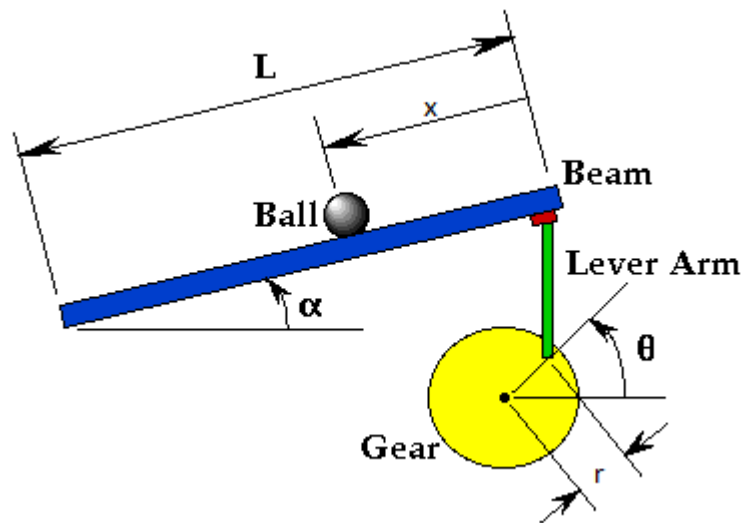


Fuente: Ball and beam Quanser. [en línea] Ball and Beam. Estados Unidos. [Consultado el 19 de noviembre de 2012] http://www.quanser.com/english/html/products/fs_product_challenge.asp?lang_code=english&pcat_code=exp-rot&prod_code=R2-ballbeam

Este sistema tiene dos lazos independientes, el externo que se encarga de controlar la posición de la bola sobre la barra por medio de la manipulación del ángulo de inclinación de esta (x/α) y un lazo interno que se encarga del control de la posición del servomotor a un ángulo determinado (θ/V). Para determinar la posición del servomotor se modela como una ganancia que varía de acuerdo al valor tomado por el ángulo α en la siguiente relación (Ecuación 14).

$$\theta = \frac{L}{r} \alpha \quad (14)$$

Figura 12. Bosquejo sistema Ball and Beam



Fuente: Bosquejo sistema bola viga [en línea] Sistema de control bola y viga. Puerto Rico. [Consultado el 19 de noviembre de 2012]
<http://ballandbeamlosli.tripod.com/>

Para el experimento de bola y de la viga se tiene el radio de la rueda de salida r , g la gravedad, la longitud L de la viga, y el ángulo de la servo DC como θ . La función de transferencia que representa el comportamiento de la planta dada por el fabricante es (Ecuación 15):

$$\frac{X(s)}{\alpha(s)} = \frac{49}{7s^2} \quad (15)$$

Los valores de los parámetros son $g=9.8\text{m/s}^2$, $L=43,18\text{cm}$ y $r=2.54\text{cm}$.

6.2. DISEÑO DE LOS CONTROLADORES DIFUSOS

Para el diseño del controlador difuso se debe definir el sistema de inferencia que se va a utilizar, las entradas y salidas del controlador, se debe establecer las funciones de pertenencia y definir las reglas.

El sistema de inferencia difuso que se va utilizar es de tipo sugeno, debido a su simplicidad a la hora de implementar, ya que no se debe llevar a cabo el proceso

de defuzzificación, además los consecuentes de las reglas ya no son conjuntos difusos sino que están representados por un valor numérico.

Las entradas del controlador difuso son la señal de error y su derivada, estas pasan por un mecanismo de inferencia generando la acción de control. Tanto la señal de error como la derivada puede ser positiva, cero o negativa (etiquetas lingüísticas). El error nos da información acerca de que tan lejos o cerca está la referencia actual a la referencia deseada y la derivada del error nos proporciona información que tan rápido o lento cambia la referencia. Si el error es negativo significa que la referencia actual está por debajo de la referencia deseada y una derivada negativa significa que está cada vez más alejada de la referencia, mientras que si se tuviera una derivada del error positiva esta significaría que la referencia actual se está acercando a la referencia deseada. Si el error y la derivada son cero esto se interpreta que la referencia esta quieta.

La salida del sistema depende según del sistema que se esté controlando. Si la salida es $u(t)$ es un controlador tipo posición y si la salida es du/dt es un controlador incremental. En el control difuso si un controlador genera la señal $u(t)$ a partir de las entradas $e(t)$ y de/dt se le denomina controlador PD difuso y si genera la señal du/dt a partir de la entradas es un controlador PI difuso [3].

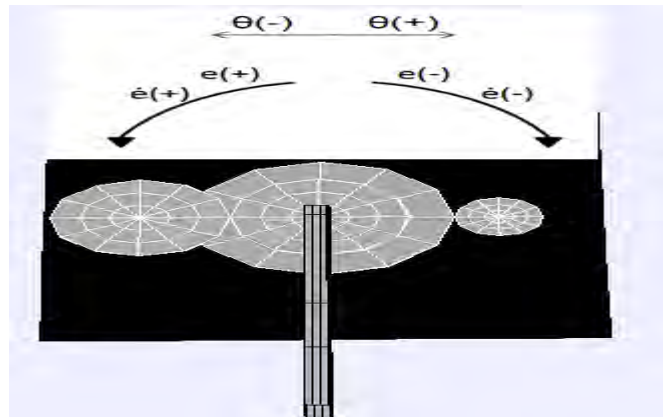
La función de pertenencia es una forma de representar gráficamente un conjunto difuso. Esta da información acerca del grado de pertenencia de un elemento a un conjunto, es decir es un valor numérico en un intervalo $[0,1]$ con el cual se expresa la medida en que un elemento cumple un determinado predicado. Para determinar una función de pertenencia, por lo general se escogen funciones sencillas para que los cálculos no sean complicados como los son las triangulares, trapezoidales y gaussianas, además el uso de funciones complejas no aporta mayor precisión.

Las reglas se definen para determinar el comportamiento de la planta. La combinación del error y la derivada del error (etiquetas lingüísticas) arrojan como resultado las posibles acciones de control sobre la planta. El valor deseado de la acción de control se pone en la intersección de la fila (error) y columna (derivada error), se usa etiquetas lingüísticas para esta variable.

6.2.1. Controlador difuso flexible link. Como se mencionó anteriormente el sistema de inferencia que se utilizara será el tipo sugeno, como se va a controlar la posición la salida del sistema de inferencia es la acción de control sencilla.

Con el esquema de la Figura 13 se puede analizar el comportamiento del flexible link, con este se puede determinar cuáles son las direcciones de aumento o decremento de las entradas, esto facilita la construcción de las reglas (Cuadro 1).

Figura 13. Comportamiento del error y su derivada en el Flexible Link



En la Figura 13 se observa que dependiendo de la rotación del motor es posible determinar cuál es el comportamiento del error y la derivada del error en cada instante de tiempo, esto nos permite estipular cual debe ser la acción de control que se debe realizar (Cuadro 1 Reglas del controlador difuso). A continuación se explica con un ejemplo de funcionamiento como es el comportamiento del controlador difuso dependiendo del estado en el que se encuentren sus dos variables de entrada.

Partiendo de un estado de reposo del módulo, o sea setpoint en 0° , una rotación en sentido horario genera un error negativo y una rotación en sentido anti horario genera un error positivo. La derivada del error en este caso permite determinar la tendencia del movimiento en un instante de tiempo futuro partiendo de lo que se sabe que está pasando con el error actual y con el error de muestras pasadas, por lo antes dicho, en la figura se observa que si nos alejamos de la referencia en sentido horario la derivada del error va a ser negativa siempre y cuando el modulo se siga moviendo en ese sentido, por el contrario si nos alejamos de la referencia en sentido anti horario la derivada del error va ser positiva siempre y cuando la rotación del motor continúe en dicho sentido.

En la Figura 14 y 15, se observa las entradas del controlador difuso donde se establecieron las siguientes etiquetas lingüísticas. MN: Muy negativo, N: Negativo, C: cero, P: Positivo, MP: Muy positivo. Los valores del error varía entre -0.7 y 0.7 que corresponde a los valores en radianes de los grados que se mueve el modulo (-45° y 45°)

Figura 14. Error Flexible Link

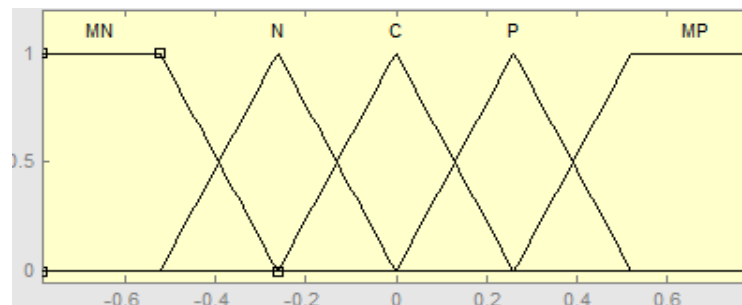
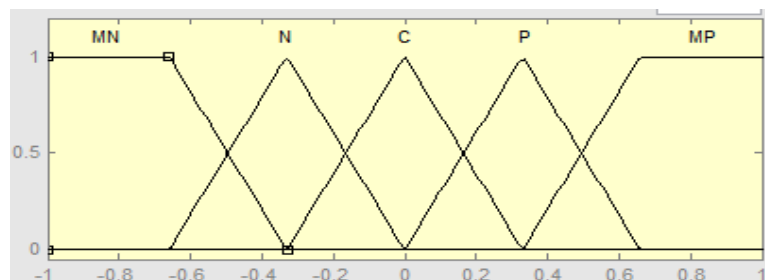
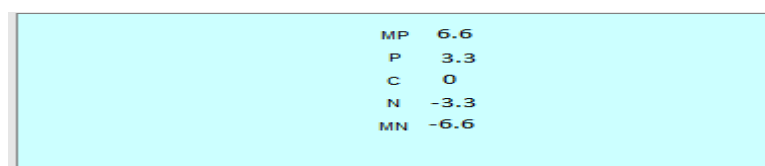


Figura 15. Derivada del error Flexible Link



En la Figura 16 se puede apreciar la salida del controlador difuso, se establecieron las siguiente etiquetas lingüísticas MN: Muy negativo, N: Negativo, C: cero, P: Positivo, MP: Muy positivo. Los valores varían entre -6.6 y 6.6 voltios, que corresponde al voltaje que se le envía al servomotor para que se mueva a la posición deseada.

Figura 16. Salida Controlador Difuso Flexible Link



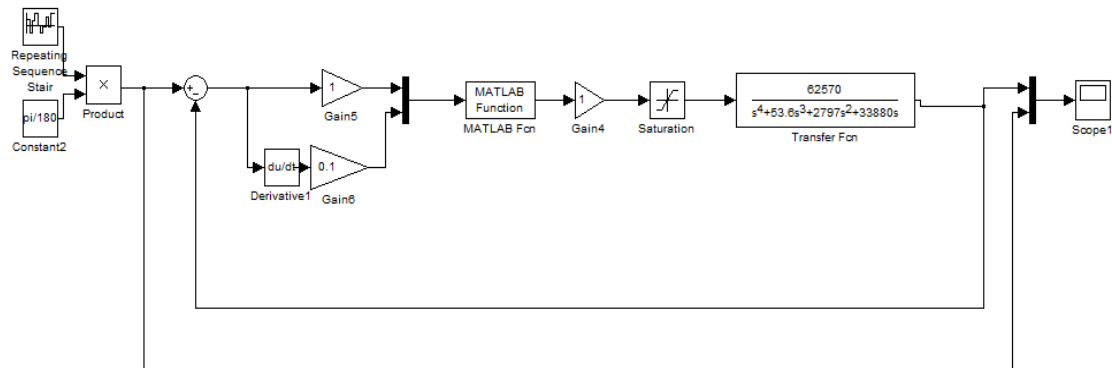
En el Cuadro 1 se puede observar las reglas del controlador difuso para el Flexible Link, esta determinan la salida del controlador y la acción de control que debe realizar.

Cuadro 1. Reglas controlador difuso Flexible link

Voltaje		Derivada del error				
		MN	N	C	P	MP
Error	MN	MN	MN	N	N	C
	N	MN	N	N	N	C
	C	N	N	C	P	P
	P	C	P	P	P	MP
	MP	C	P	P	MP	MP

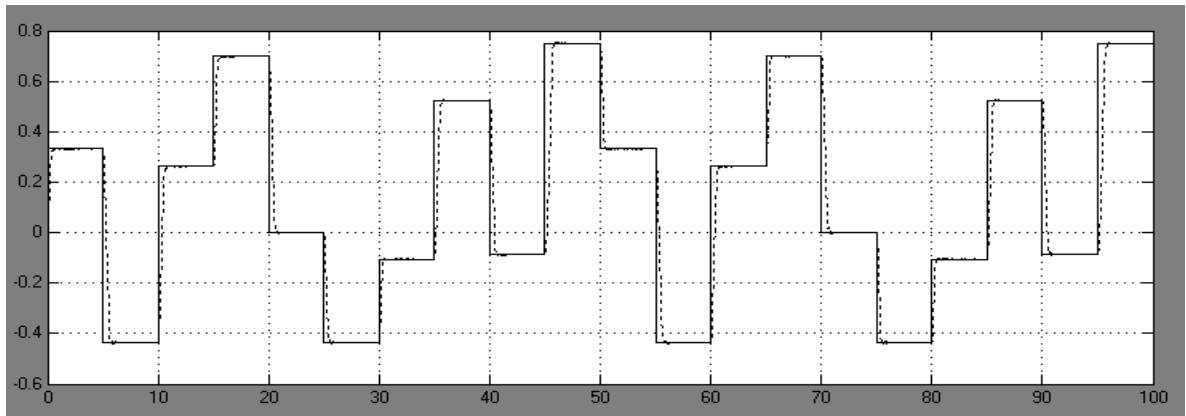
El diagrama de bloques de la Figura 17 corresponde al controlador difuso implementado en simulink, con el cual se pretende observar la salida del controlador con el Flexible Link.

Figura 17. Diagrama bloques controlador difuso Flexible Link



La entrada para simulación fue una posición deseada para que el Flexible Link llegara a esta, como la salida del controlador sigue la referencia (posición deseada) se concluye que el controlador difuso diseñado funciona de forma adecuada, en la Figura 18 se observa este comportamiento (línea solida referencia, línea punteada salida controlador).

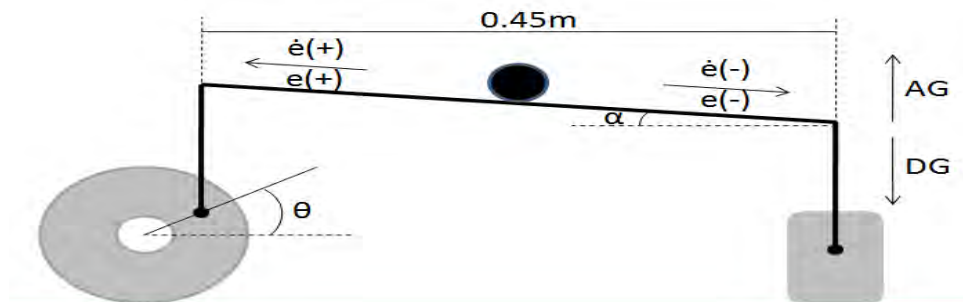
Figura 18. Salida del Flexible Link con el controlador difuso



6.2.2. Controlador difuso Ball and Beam. En esta parte se describe el controlador difuso para este módulo teniendo cinco conjuntos difusos tanto para las entradas y la salida, la acción de control es no incremental.

En la Figura 19 se observa el esquema de cómo debe variar el error y su derivada para poder construir de manera más fácil las reglas del controlador (Cuadro 2).

Figura 19. Comportamiento del error y su derivada para el Ball and Beam.



En la Figura 19 se observa que la rotación del motor genera una inclinación (ángulo θ) de la viga y por consiguiente un movimiento del balón a lo largo de ella, para este módulo una rotación en sentido horario genera una disminución de la inclinación y una rotación en sentido anti horario genera un aumento, partiendo de esto es posible determinar cuál es el comportamiento del error y la derivada del error en cada instante de tiempo, y al igual que en los otros dos módulos esto nos permite estipular cual debe ser la acción de control que se debe realizar (Cuadro 2 Reglas del controlador difuso).

Si partimos un estado de equilibrio del módulo, por ejemplo setpoint en 0.22m, una rotación en sentido horario genera un error positivo y una rotación en sentido anti horario genera un error negativo. La derivada del error del ball and beam le ayuda al controlador a estimar el movimiento futuro del balón sabiendo cuál es su sentido de desplazamiento sobre la viga cuando el motor está variando su sentido de giro, por consiguiente si el desplazamiento del balón es hacia la derecha es porque el motor va en sentido anti horario y tendremos una derivada negativa, por el contrario cuando el balón va hacia la izquierda del punto de referencia es porque el motor va en sentido horario lo que genera una derivada positiva.

En la Figura 20 y 21, se observa las entradas del controlador difuso donde se establecieron las siguientes etiquetas lingüísticas. MN: Muy negativo, N: Negativo, C: cero, P: Positivo, MP: Muy positivo. Los valores del error varían entre -0.45 m y 0.45 m ya que son los valores máximos que se pueden generar al restar la entrada deseada a la planta y la posición real de la bola.

Figura 20. Error Ball and Beam

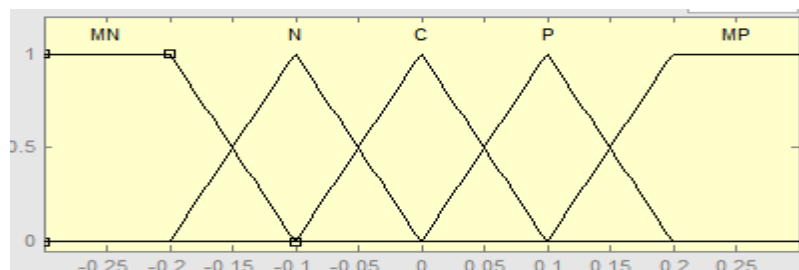
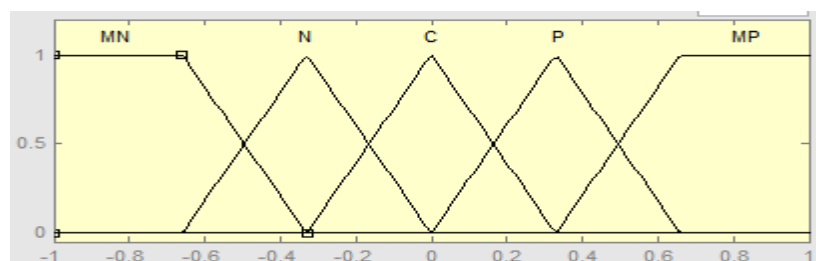


Figura 21. Derivada del error Ball and Beam



En la Figura 22 se puede apreciar la salida del controlador difuso, se establecieron las siguiente etiquetas lingüísticas DGR: Disminuir grados rápido, DG: Disminuir grados, Q: No actuar, AG: Aumentar grados, AGR: Aumentar grados rápidamente. Los valores varían entre -0.047 y 0.047 radianes, que corresponde al ángulo que varía la barra.

Figura 22. Salida controlador difuso Ball and beam

AGR	0.03
AG	0.015
Q	0
DG	-0.015
DGR	-0.03

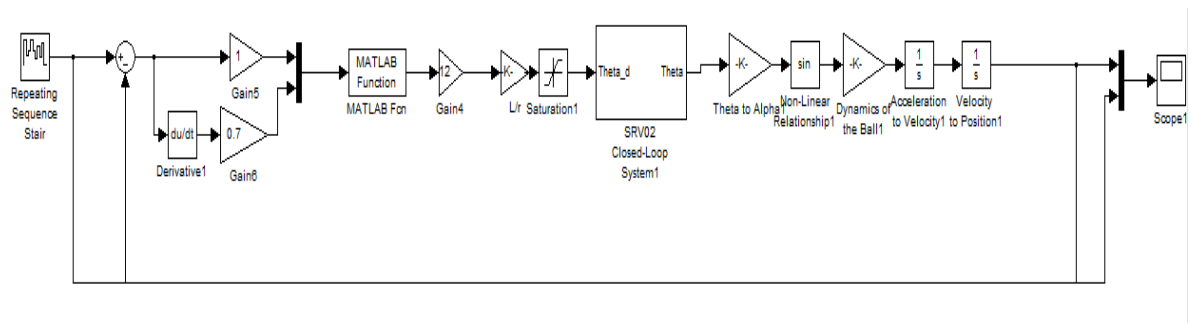
En el Cuadro 2 se puede observar las reglas del controlador difuso para el Ball and Beam, estas determinan la salida del controlador y la acción de control que debe realizar.

Cuadro 2. Reglas controlador difuso Ball and Beam

α		Derivada del error				
		MN	N	C	P	MP
Error	MN	DGR	DGR	DG	DG	Q
	N	DGR	DG	DG	Q	Q
	C	DGR	DG	Q	AG	AGR
	P	Q	Q	AG	AG	AGR
	MP	Q	AG	AG	AGR	AGR

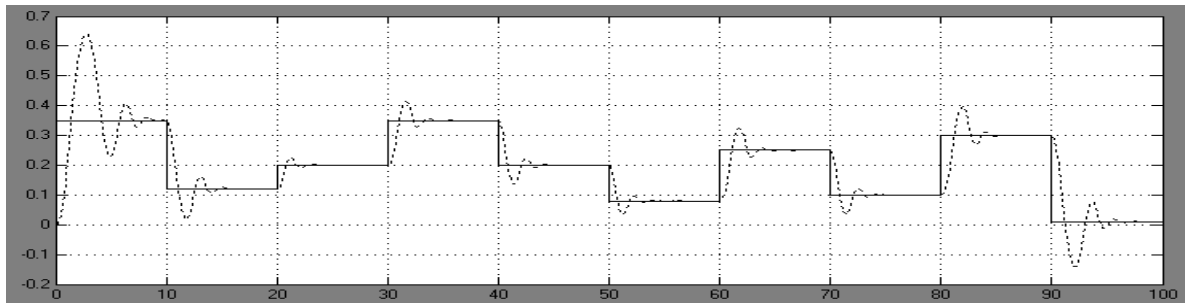
En la Figura 23 se observa el diagrama de bloques del controlador difuso implementando en simulink, con el cual se pretende observar la salida del controlador con el Ball and beam.

Figura 23. Diagrama bloques controlador difuso Ball and Beam



La entrada para simulación fue una posición deseada en metros para que la bola llegara a esa posición en la barra, el controlador difuso arroja el ángulo α que se debe mover la barra, luego mediante la relación previamente dada en la ecuación 3 se halla el ángulo θ que debe moverse el servomotor, como la salida del controlador sigue la referencia (posición deseada) se concluye que el controlador difuso diseñado funciona de forma adecuada, en la Figura 24 se observa este comportamiento (línea sólida referencia, línea punteada salida controlador).

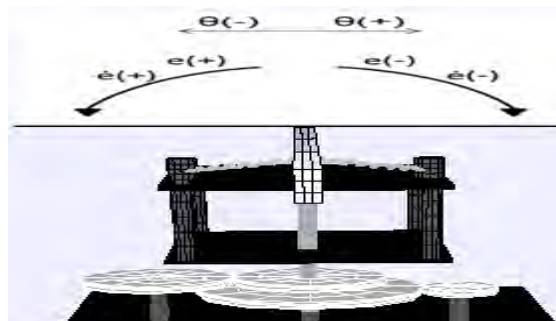
Figura 24. Salida del Ball and Beam con el controlador difuso



6.2.3. Controlador difuso Flexible Joint. El Flexible Joint tiene el mismo funcionamiento que el Flexible Link, es por eso que sus entradas, salida y reglas no cambian.

Con el esquema de la Figura 25 se puede identificar las direcciones de aumento o decremento de las entradas para facilitar la construcción de las reglas (Cuadro 3).

Figura 25. Comportamiento del error y su derivada en el Flexible Joint



El comportamiento del error y la derivada del error del Flexible Joint es igual al comportamiento de estas entradas en el Flexible Link, debido a que tienen el mismo comportamiento.

En la Figura 26 y 27, se observa las entradas del controlador difuso donde se establecieron las siguientes etiquetas lingüísticas. MN: Muy negativo, N: Negativo, C: cero, P: Positivo, MP: Muy positivo. Los valores del error varía entre -0.7 y 0.7 que corresponde a los valores en radianes de los grados que se mueve el modulo (-45° y 45°)

Figura 26. Error Flexible Joint

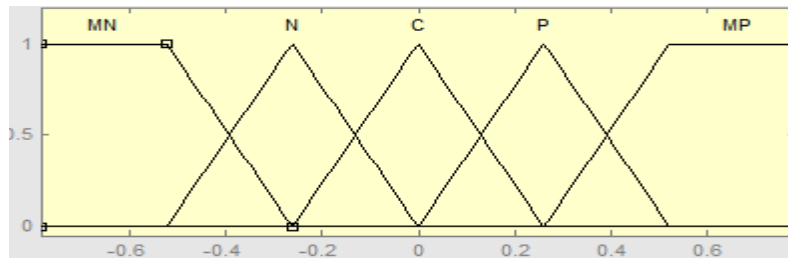
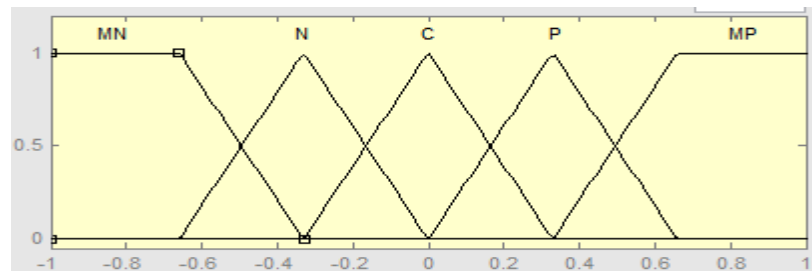
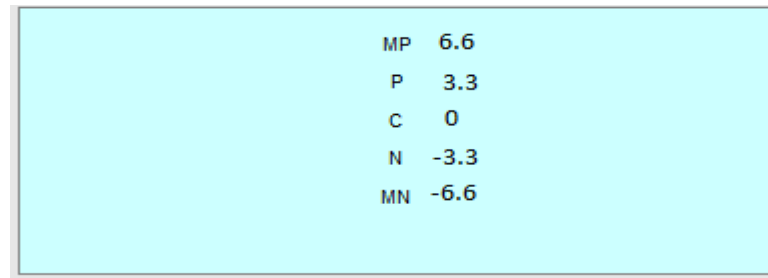


Figura 27. Derivada del error Flexible Joint



En la Figura 28 se puede apreciar la salida del controlador difuso, se establecieron las siguiente etiquetas lingüísticas MN: Muy negativo, N: Negativo, C: cero, P: Positivo, MP: Muy positivo. Los valores varían entre -6.6 y 6.6 voltios, que corresponde al voltaje que se le envía al servomotor para que se mueva a la posición deseada.

Figura 28. Salida Controlador Difuso Flexible Joint



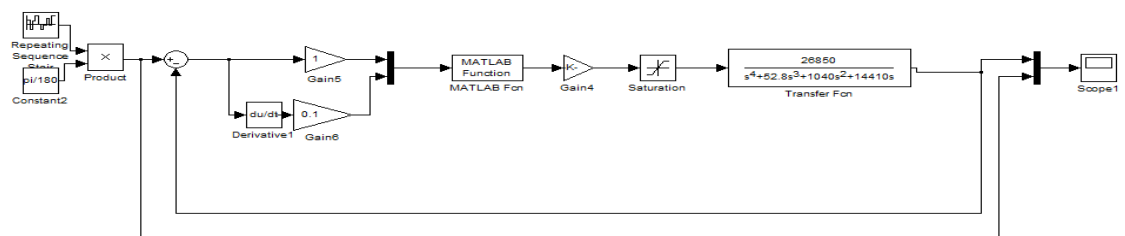
En el Cuadro 3 se puede observar las reglas del controlador difuso para el Flexible Joint, esta determinan la salida del controlador y la acción de control que debe realizar.

Cuadro 3. Reglas controlador difuso Flexible Joint

Voltaje		Derivada del error				
		MN	N	C	P	MP
Error	MN	MN	MN	N	N	C
	N	MN	N	N	N	C
	C	N	N	C	P	P
	P	C	P	P	P	MP
	MP	C	P	P	MP	MP

El diagrama de bloques de la Figura 29 corresponde al controlador difuso implementado en simulink, con el cual se pretende observar la salida del controlador con el Flexible Joint.

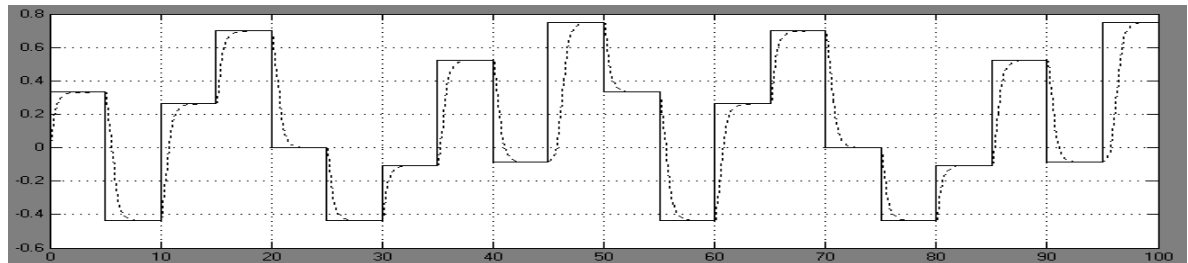
Figura 29. Diagrama bloques controlador difuso Flexible Joint



La entrada para simulación fue una posición deseada para que el Flexible Joint llegara a esta, como la salida del controlador sigue la referencia (posición deseada) se concluye que el controlador difuso diseñado funciona de forma

adecuada, en la Figura 30 se observa este comportamiento (línea solida referencia, línea punteada salida controlador).

Figura 30. Salida del Flexible Joint con el controlador difuso



6.3. DISEÑO CONTROLADORES PID

El controlador PID (Proporcional, Integral y Derivativo) reúne las ventajas de cada de las tres acción de control individuales. La acción proporcional aumenta la rapidez de respuesta del sistema pero tiene error en estado estacionario, este error lo elimina la acción integral pero la oscilación es mayor, esto se corrige mediante la acción derivativa y aumenta la estabilidad.

Se diseñara un controlador PID paralelo (Ecuación 16), para esto se deben hallar los valor de las ganancias de cada una de las acciones de controla (Proporcional, Integral y Derivativo), para esto se usara el método analítico, el método de oscilación de ziegler y nichols y ajuste en linea.

$$G_c = K_p + K_d s + \frac{K_i}{s} \quad (16)$$

6.3.1. Controlador PID Flexible Joint

6.3.1.1. Método analítico. En el método analítico se diseñara un controlador PD, como la planta es de cuarto orden se lleva a un segundo orden (Ecuación 17) (ver Anexo A) para mayor facilidad en el diseño analítico.

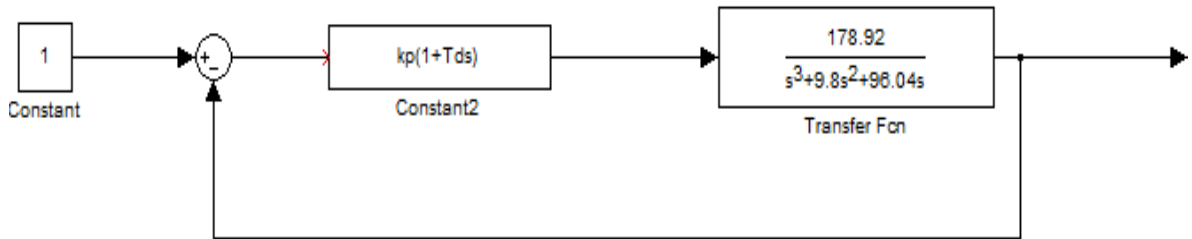
$$G(s) = \frac{178.92}{s^2 + 9.8s + 96.04} \quad (17)$$

Para el diseño analítico se usara la función de transferencia de segundo orden multiplicado por el integrador puro que se retiró previamente en el Anexo A, quedando como resultado una función de transferencia de orden 3 (Ecuación 18)

$$G(s) = \frac{178.92}{s^3 + 9.8s^2 + 96.04s} \quad (18)$$

Se desarrolla el siguiente sistema (Figura 31) en lazo cerrado para hallar los valores de la acción proporcional y la acción derivativa.

Figura 31. Control PD Flexible Joint



Al desarrollarse el sistema en lazo cerrado se obtiene la ecuación 19 que al igualarla con la ecuación característica se pueden hallar los valores del controlador PD.

$$s^3 + 9.8s^2 + (96.04 + 178.92KpTd)s + 178.92Kp \quad (19)$$

La ecuación característica se halla con los siguientes parámetros (Ecuación 20):

$$\epsilon = 1.4 \quad Ts = 2 \quad Wn = 1.2 \quad (20)$$

Dando como resultado la siguiente ecuación 21

$$s^3 + 9.8s^2 + 25.43s + 11.85 \quad (21)$$

Al igualar la ecuación 19 y 21, se obtiene (Ecuación 22):

$$K_p = 0.06 \quad K_d = -0.39 \quad (22)$$

A pesar que la ganancia derivativa dio negativa, esta funciona correctamente cuando se simularon estos valores.

La simulación y respuesta del controlador PD para el flexible joint se observa en la Figura 32 y 33 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 32. Diagrama de bloques PD método analítico Flexible Joint

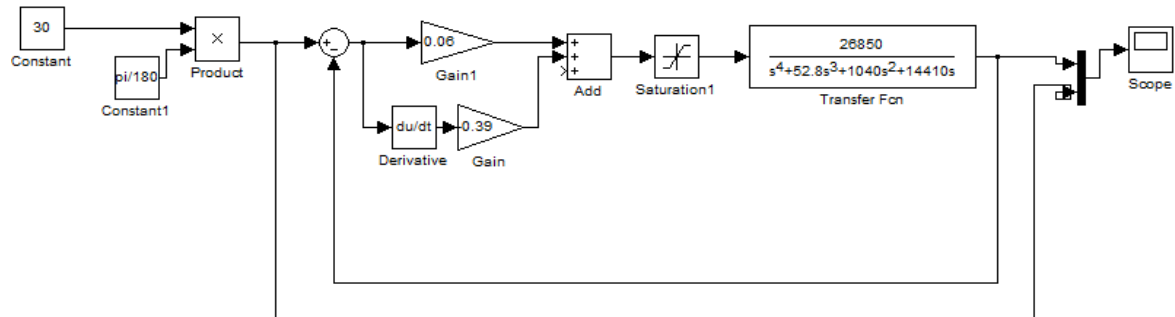
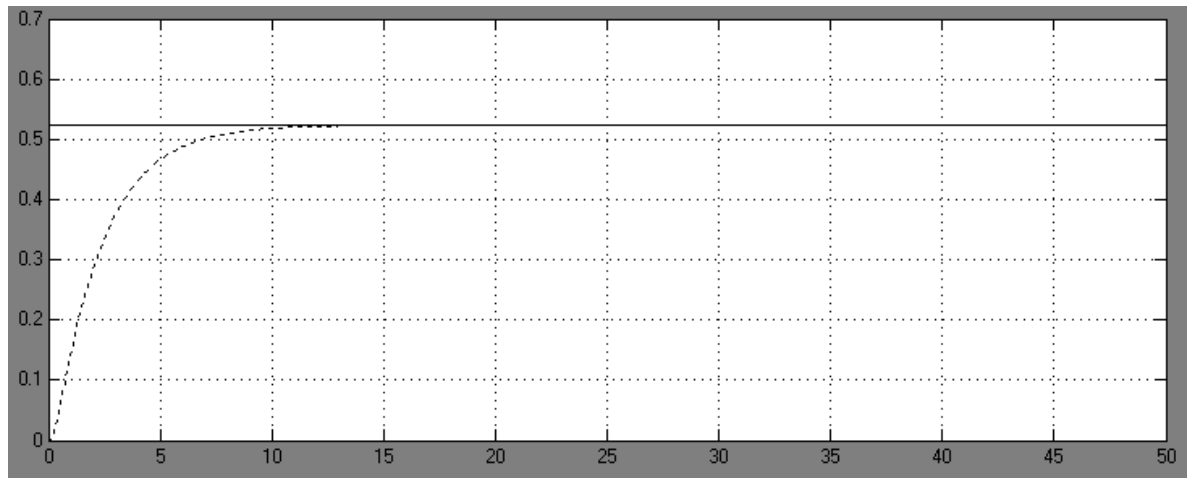
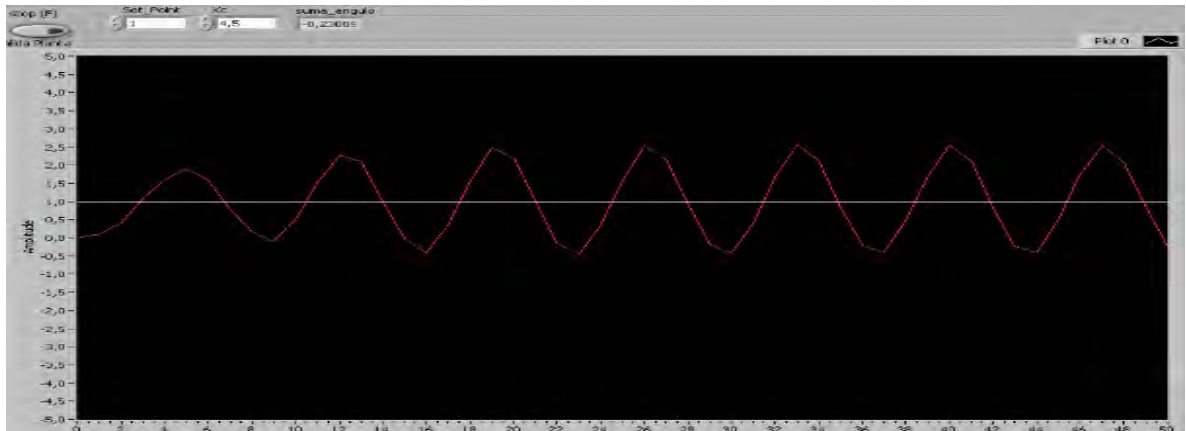


Figura 33. Respuesta método analítico Flexible Joint



6.3.1.2. Método empírico. Con el método empírico de ziegler y nichols se ajustaran los valores del controlador PID por medio del método de la oscilación sostenida. Como primer paso se empezó a variar una constante K_c (Ganancia crítica) hasta que el flexible joint empezara a oscilar, ya oscilando el módulo se determinó el T_c (periodo de oscilación), como se observa en la Figura 34.

Figura 34. Método Ziegler y Nichols Flexible Joint



Se obtuvieron los siguientes valores (Ecuación 23):

$$K_c = 4.5 \quad T_c = 0.6 \quad (23)$$

Estos valores se reemplazaron según a los valores del Cuadro 4 determinada por ziegler y nichols para el método de la oscilación sostenida.

Cuadro 4. Parámetros controladores PID según el método de la oscilación de Ziegler y Nichols

TIPO DE PID	K_p	T_i	T_d
P	$0.5K_c$	∞	0
PI	$0.45K_c$	$T_c/1.2$	0
PID	$0.6K_c$ o $1K_c$	$T_c/2$	$T_c/8$

Al reemplazar los valores se obtienen el K_p , el T_i y el T_d , y con estos valores se hallan los valores del PID (Ecuación 25), donde

$$K_p = K_p \quad K_i = \frac{K_p}{T_i} \quad K_d = K_p * T_d \quad (24)$$

$$K_p = 4.5 \quad K_i = 15 \quad K_d = 0.33 \quad (25)$$

La simulación y respuesta del controlador PID por el método de ziegler y nichols para el flexible joint se encuentra en la Figura 35 y 36 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 35. Diagrama de bloques PID método ziegler y nichols Flexible Joint

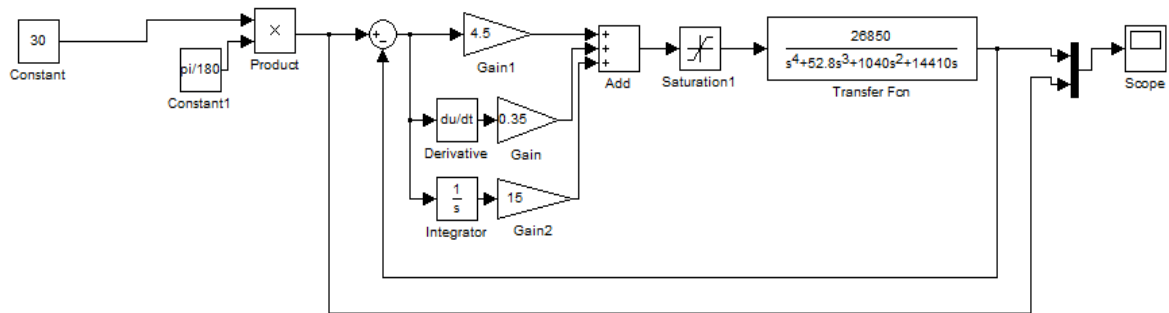
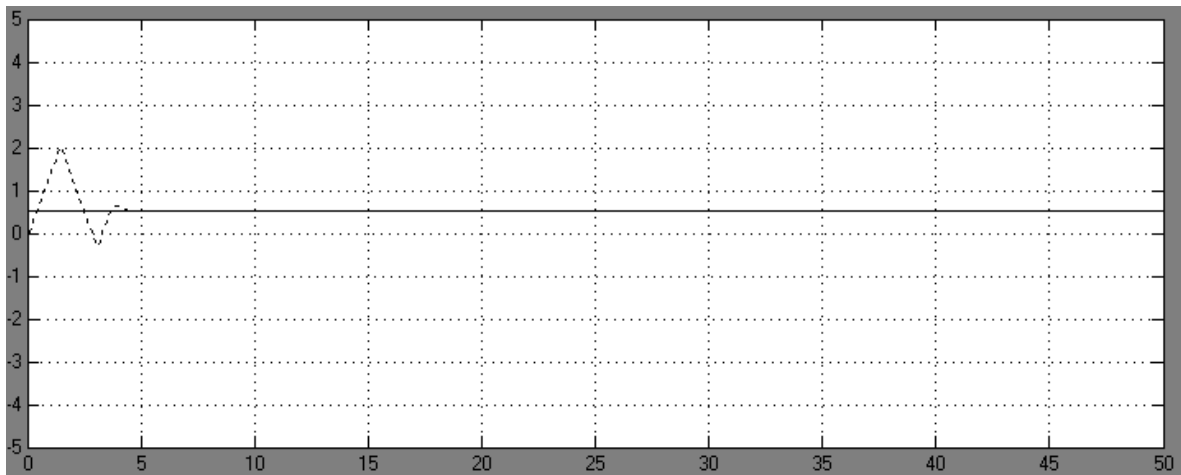
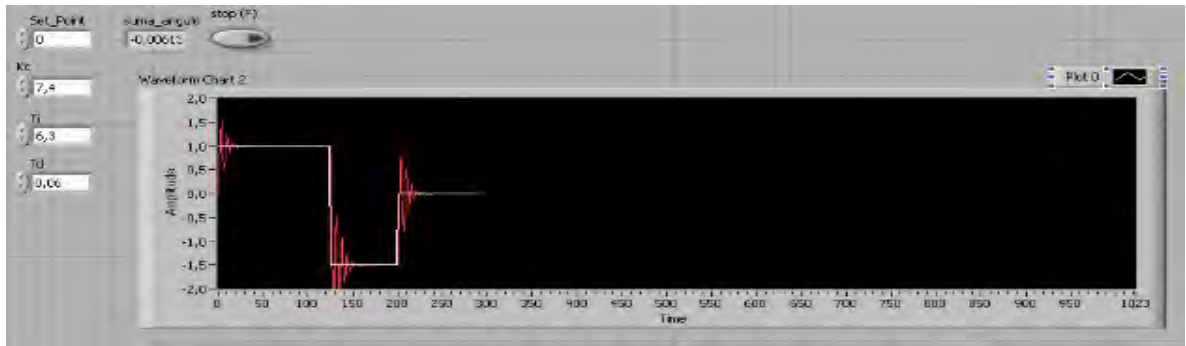


Figura 36. Respuesta método ziegler y nichols Flexible Joint



Tomando como punto de partida las ganancias obtenidas con ziegler y nichols se ajustan las ganancias para obtener una respuesta lo más satisfactoria posible del sistema controlado (Figura 37).

Figura 37. Método prueba y error Flexible Joint



Se obtuvieron los siguientes resultados (Ecuación 27):

$$Kp = Kc * 0.6 \quad Ki = \frac{Kp}{Ti} \quad kd = Kp * Td \quad (26)$$

$$Kp = 4.4 \quad ki = 0.7 \quad kd = 0.26 \quad (27)$$

La simulación y respuesta del controlador PID por el ajuste de ganancias para el flexible joint se encuentra en las Figuras 38 y 39 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 38. Diagrama de bloques PID ajuste de ganancias Flexible Joint

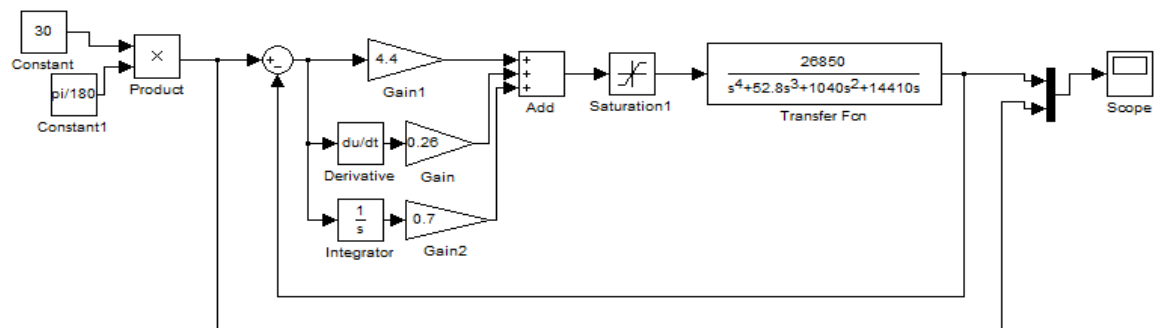
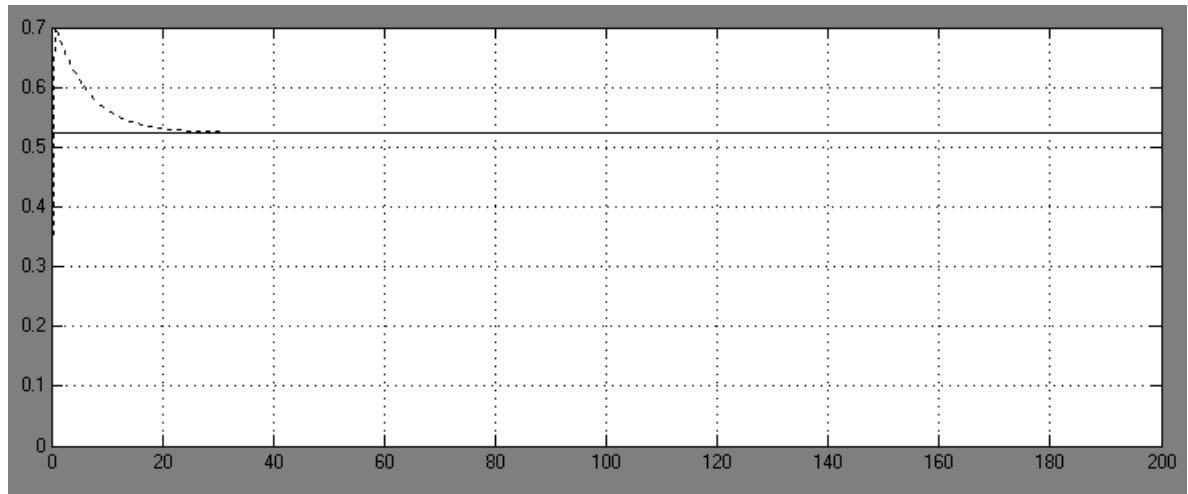


Figura 39. Respuesta ajuste de ganancias Flexible Joint



En el Cuadro 5 se encuentran consignados los valores obtenidos por los tres métodos para el Flexible Joint.

Cuadro 5. Valores PID Flexible Joint

MÉTODO	Kp	Ki	Kd
Analítico	0.06		-0.39
Ziegler y Nichols	4.5	15	0.33
Ajuste en línea	4.4	0.7	0.26

6.3.2. Controlador PID Flexible Link

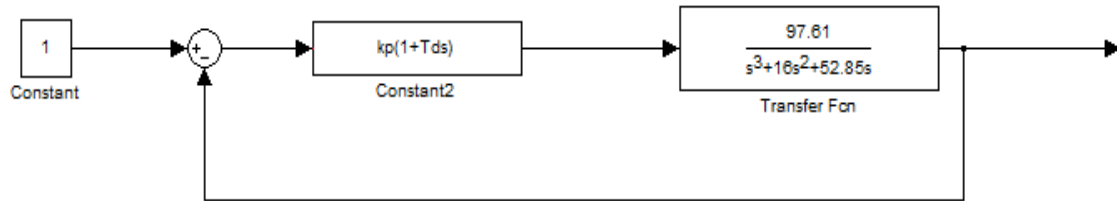
6.3.2.1. Método analítico. Para el diseño por el medio analítico se usa la función de transferencia del flexible link de orden 2 (Ecuación 28) previamente hallada en el Anexo B y se multiplica por el integrador puro que se retiró cuando se pasó a orden 2 (Ecuación 29).

$$G(s) = \frac{97.61}{s^2 + 16s + 52.85} \quad (28)$$

$$G(s) = \frac{97.61}{s^3 + 16s^2 + 52.85s} \quad (29)$$

Se desarrolla el siguiente sistema (Figura 34) en lazo cerrado para hallar los valores de la acción proporcional y la acción derivativa.

Figura 40. Control PD Flexible Link



Con la Ecuación 30 obtenida al cerrarse el lazo se halla los valores del controlador PD al igualarse con la ecuación característica.

$$s^3 + 16s^2 + (52.85 + 97.61KpTd)s + 97.61Kp \quad (30)$$

La ecuación característica se halla con los siguientes parámetros (Ecuación 31):

$$\epsilon = 1.6 \quad Ts = 1 \quad Wn = 2.5 \quad (31)$$

Dando como resultado la siguiente ecuación 32

$$s^3 + 16s^2 + 70.2s + 50 \quad (32)$$

Al igualar la ecuación 30 y 32, se obtiene que (Ecuación 33):

$$Kp = 0.51 \quad Kd = 0.17 \quad (33)$$

La simulación y respuesta del controlador PD para el flexible link se encuentra en las Figuras 41 y 42 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 41. Diagrama de bloques PD método analítico Flexible Link

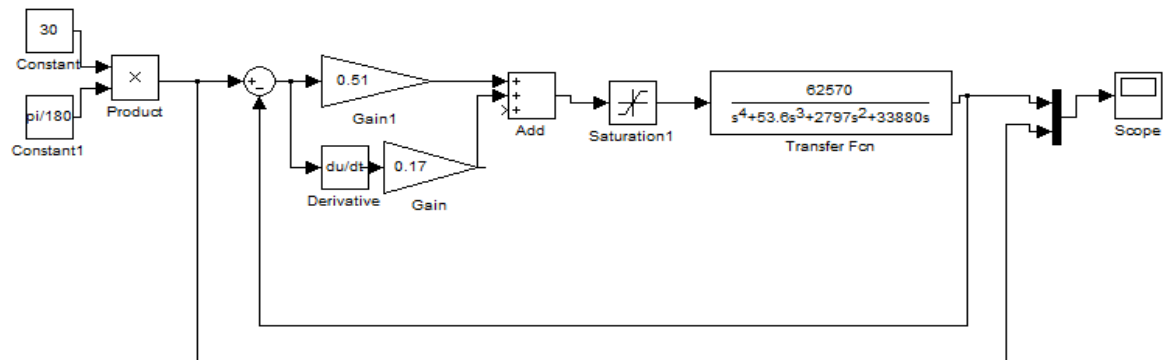
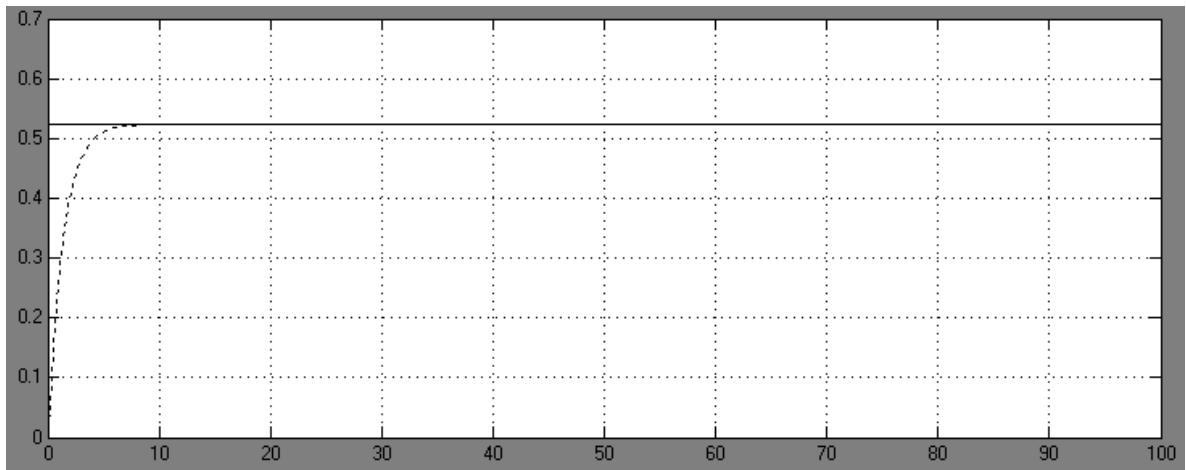
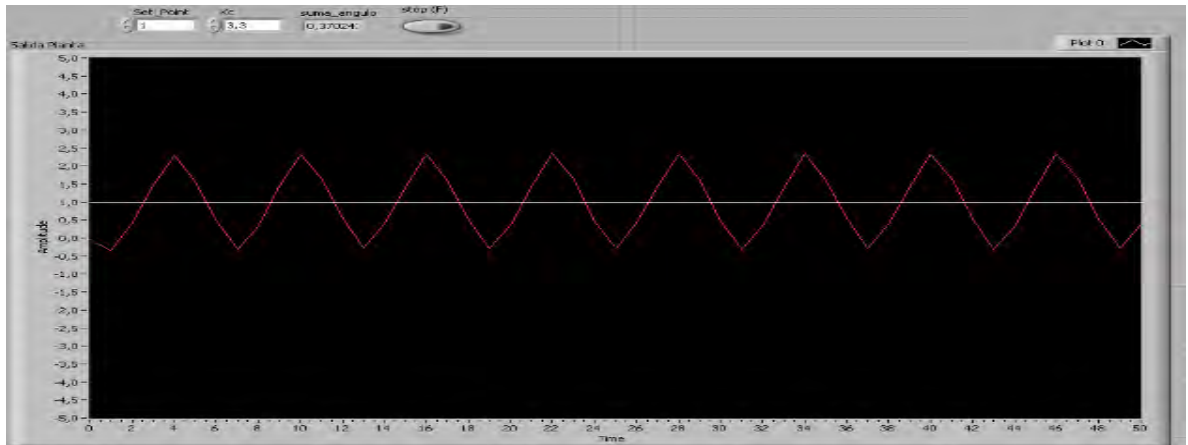


Figura 42. Respuesta método analítico Flexible Link



6.3.2.2. Método empírico. Para hallar los valores del PID por medio del método de oscilación de ziegler y nichols se empezó a variar una constante K_c hasta que el sistema empezara a oscilar, luego con el sistema oscilando se tomó el periodo de este (Figura 43).

Figura 43. Método Ziegler y Nichols Flexible Link



Se obtuvieron los siguientes valores (Ecuación 34):

$$K_c = 4.5 \quad T_c = 0.6 \quad (34)$$

Luego con ayuda de la tabla 4 se reemplazan estos valores para obtener los valores del PID (Ecuación 35), teniendo como resultado:

$$K_p = 3.3 \quad K_i = 11 \quad K_d = 0.24 \quad (35)$$

La simulación y respuesta del controlador PID por el método de ziegler y nichols para el flexible link se encuentra en las Figura 44 y 45 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 44. Diagrama de bloques PID método ziegler y nichols Flexible Link

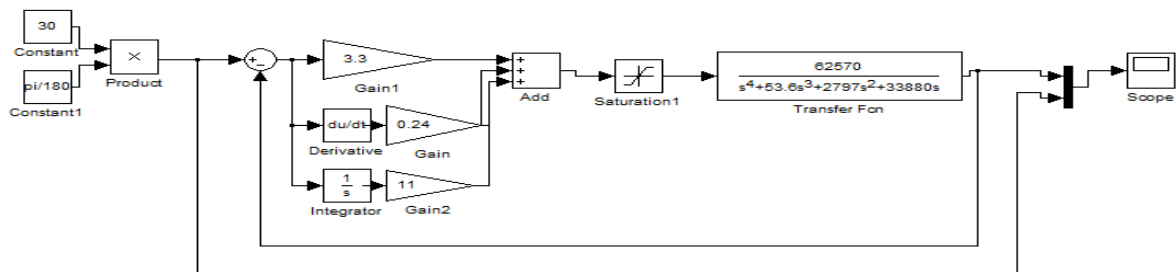
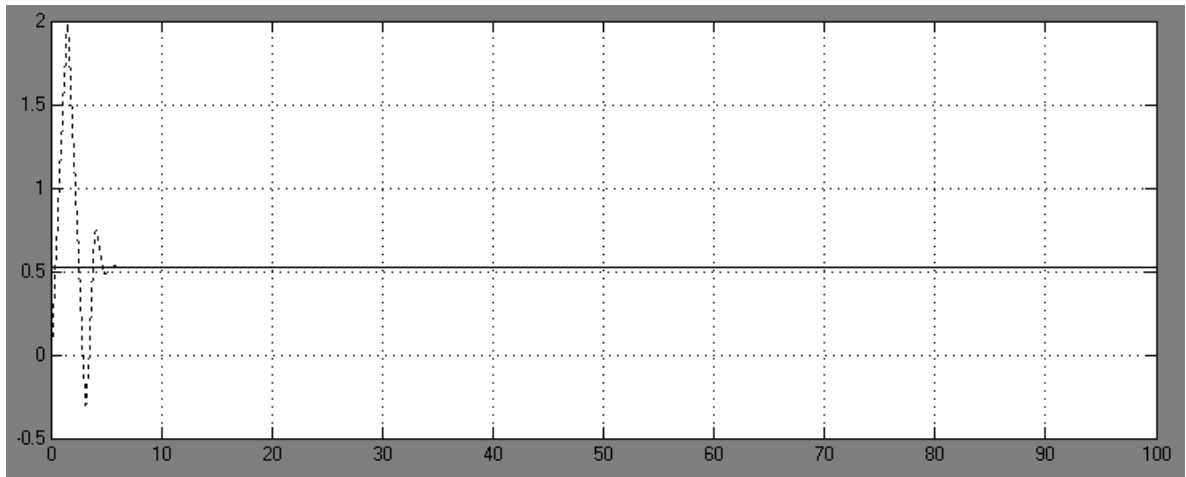
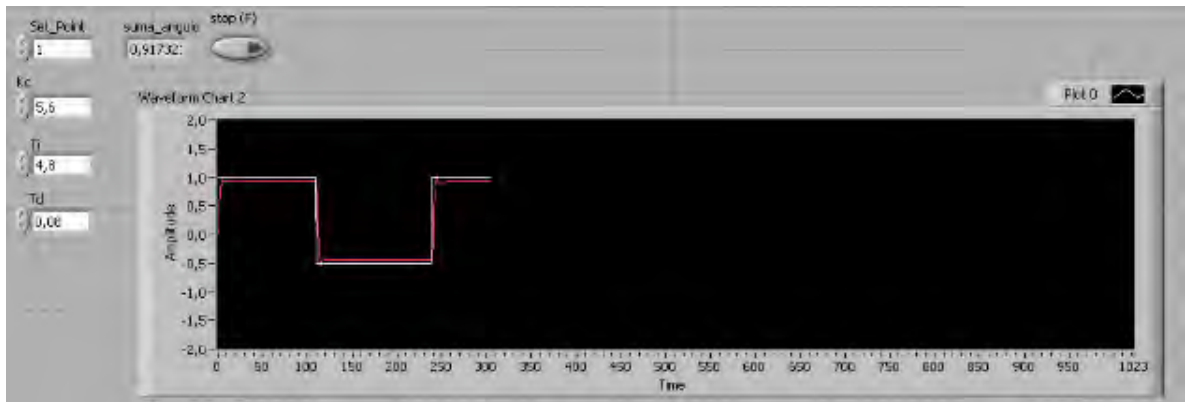


Figura 45. Respuesta método ziegler y nichols Flexible Link



Tomando como punto de partida las ganancias obtenidas con ziegler y nichols se ajustan las ganancias para obtener una respuesta lo más satisfactoria posible del sistema controlado (Figura 46).

Figura 46. Método prueba y error Flexible Link



Se obtuvieron los siguientes resultados, que al remplazarlos en la ecuación 26 se obtiene los valores del PID (Ecuación 36):

$$Kp = 3.36 \quad ki = 0.7 \quad kd = 0.26 \quad (36)$$

La simulación y respuesta del controlador PID por el ajuste de ganancias para el flexible link se encuentra en las Figuras 47 y 48 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 47. Diagrama de bloques PID ajuste de ganancias Flexible Link

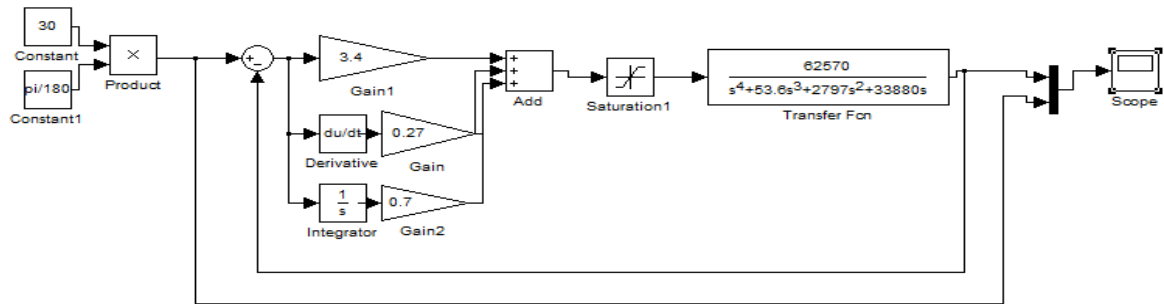
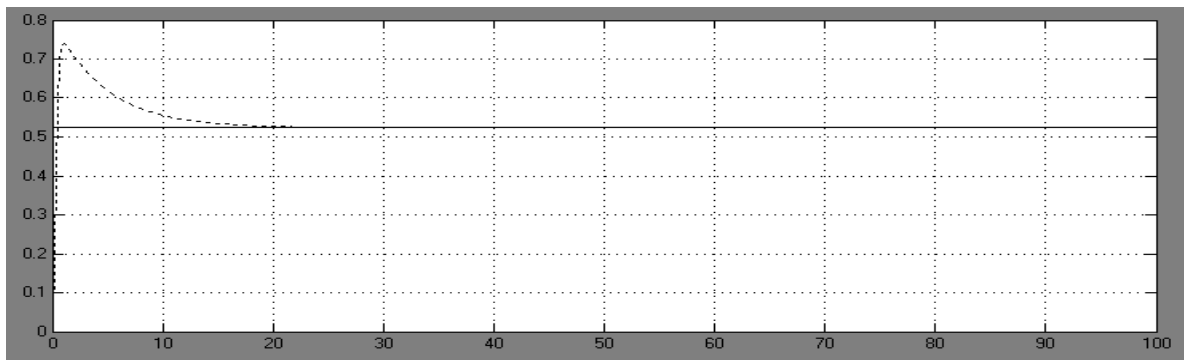


Figura 48. Respuesta ajuste de ganancias Flexible Link



En el Cuadro 6 se encuentran consignados los valores obtenidos por los tres métodos para el Flexible Link.

Cuadro 6. Valores PID Flexible Link

MÉTODO	Kp	Ki	Kd
Analítico	0.51		0.17
Ziegler y Nichols	3.3	11	0.24
Ajuste en línea	3.36	0.7	0.26

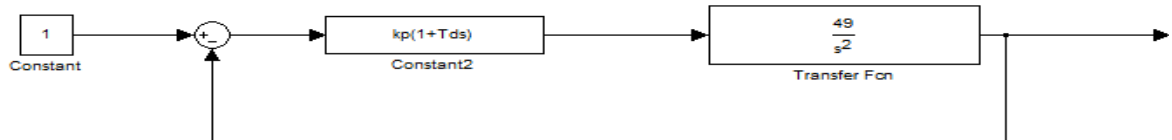
6.3.3. Controlador PID Ball and Beam

6.3.3.1. Método analítico. Para el diseño analítico del PD del Ball and Beam se usara la función de transferencia (ecuación 6).

$$\frac{X(s)}{\alpha(s)} = \frac{49}{7s^2} \quad (6)$$

Se desarrolla es sistema en lazo cerrado con el controlador (Figura 49) y se halla la ecuación para hallar los valores del controlador (Ecuación 37)

Figura 49. Controlador PD Ball and Beam



$$s^2 + 7KpTds + 7Kp \quad (37)$$

La ecuación característica se halla con los siguientes parámetros (Ecuación 38):

$$\epsilon = 1.2 \quad Ts = 8 \quad Wn = 0.41 \quad (38)$$

Dando como resultado la siguiente ecuación 39

$$s^2 + 0.98s + 0.1681 \quad (39)$$

Al igualar la ecuación 37 y 39, se obtiene que (Ecuación 40):

$$Kp = 0.02 \quad Kd = 0.14 \quad (40)$$

La simulación y respuesta del controlador PD para el ball and beam se encuentra en las Figuras 50 y 51(línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 50. Diagrama de bloques PD método analítico lazo externo Ball and Beam

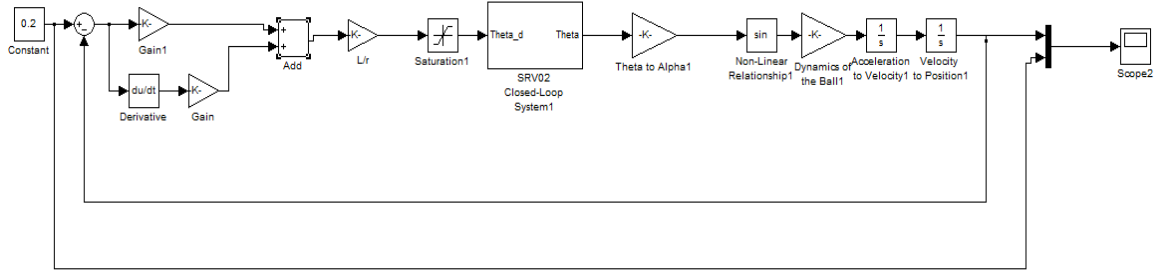
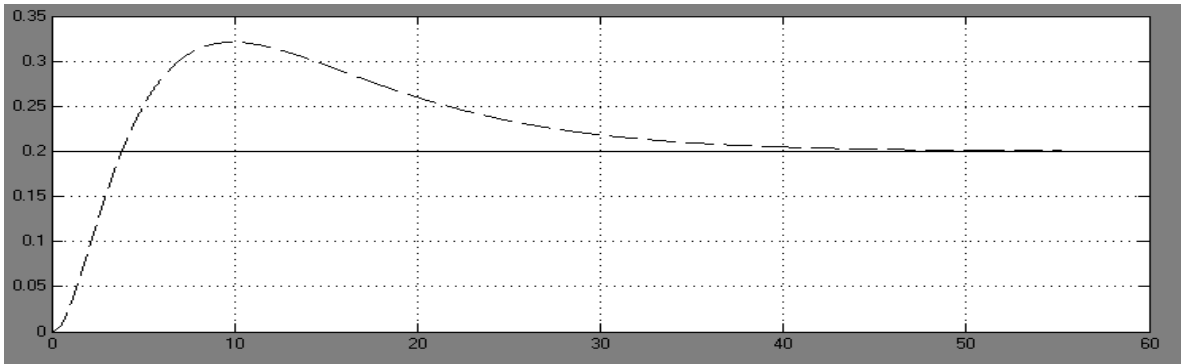


Figura 51. Respuesta método analítico lazo externo Ball and Beam

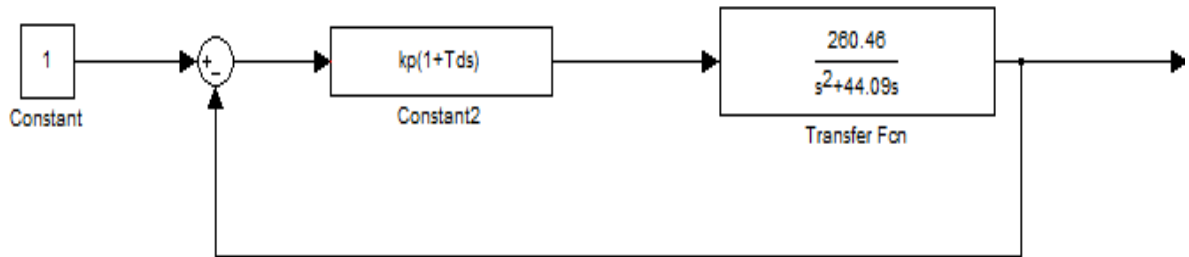


Para el ball and beam también se necesita diseñar un controlador PD para el lazo interno (servomotor), para este diseño se usara la función de transferencia del servomotor (ecuación 45).

$$\frac{\theta(s)}{V(s)} = \frac{260.46}{s^2 + 44.09s} \quad (45)$$

Se desarrolla el sistema en lazo cerrado con el controlador PD (Figura 38) para poder obtener la ecuación y así hallar sus valores (Ecuación 46).

Figura 52. Controlador PD lazo interno Ball and Beam (Servomotor)



$$s^2 + (44.09 + 260.46K_pT_d)s + 260.46K_p \quad (46)$$

La ecuación característica se halla con los siguientes parámetros (Ecuación 47):

$$\epsilon = 1.2 \quad T_s = 5 \quad W_n = 0.66 \quad (47)$$

Dando como resultado la siguiente ecuación 48

$$s^2 + 1.58s + 0.43 \quad (48)$$

Al igualar la ecuación 46 y 48, se obtiene que los valores del PD (Ecuación 49):

$$K_p = 0.00165 \quad K_d = -0.16 \quad (49)$$

La simulación y respuesta del controlador PD para el lazo interno del ball and beam (Servomotor) se encuentra en las Figuras 53 y 54 (línea solida referencia, línea punteada salida controlador) respectivamente.

Figura 53. Diagrama de bloques PD método analítico lazo interno Ball and Beam (Servomotor)

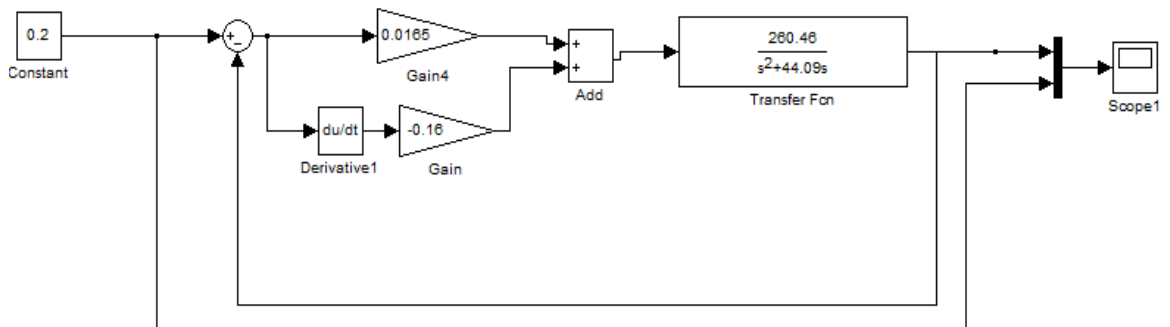
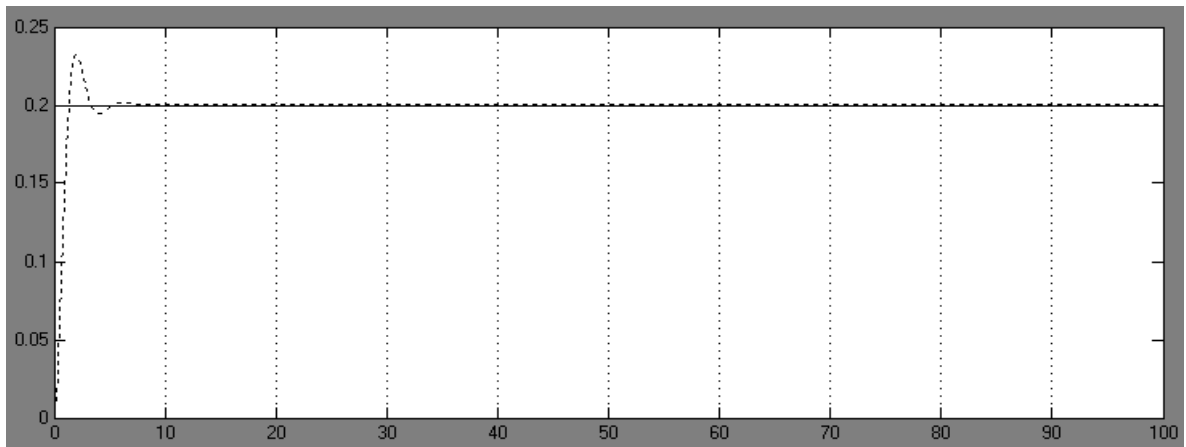


Figura 54. Respuesta método analítico lazo interno Ball and Beam (Servomotor)



6.3.3.2. Método empírico. En la Figura 55 y 56 se observan las gráficas obtenidas al utilizar el método de oscilación de ziegler y nichols, donde se hallaron los valores de la ganancia crítica y el periodo crítico.

Figura 55. Método Ziegler y Nichols lazo externo Ball and Beam

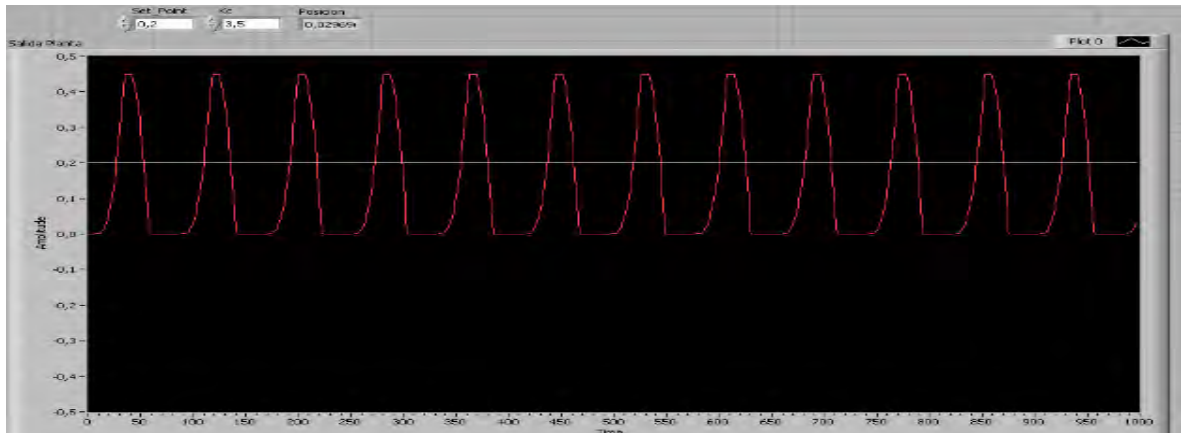
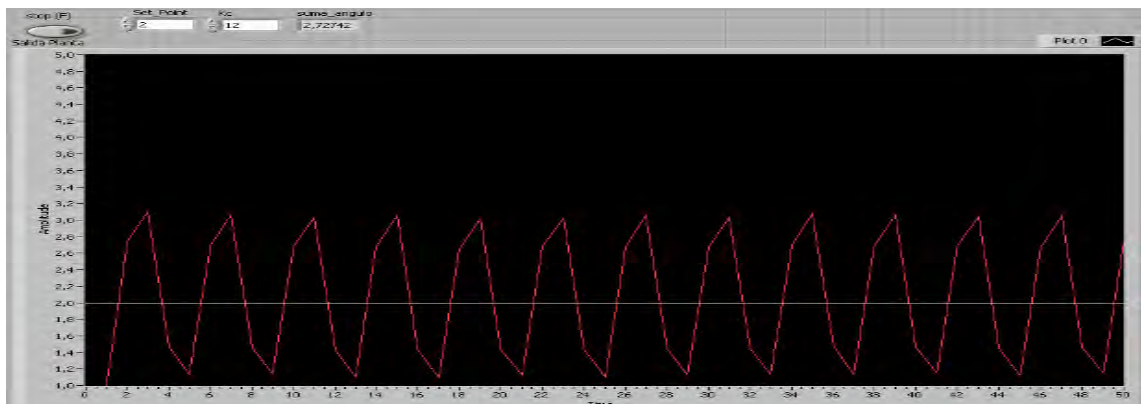


Figura 56. Método Ziegler y Nichols lazo interno Ball and Beam (Servomotor)



Se obtuvieron los siguientes valores:

$$K_c = 3.5 \quad T_c = 8 \quad (50) \text{ Lazo externo Ball and Beam}$$

$$K_c = 12 \quad T_c = 0.4 \quad (51) \text{ Lazo interno Ball and Beam (Servomotor)}$$

Al reemplazar los valores de las ecuaciones 50 y 51 en la Tabla 4, se obtienen el K_p , el T_i y el T_d , y con estos valores se hallan los valores del PID al reemplazarlos en la ecuación 24 obteniendo (Ecuación 52 y 53 respectivamente).

$$Kp = 2.1 \quad Ki = 0.525 \quad Kd = 2.1 \quad (52) \text{ Lazo externo Ball and Beam}$$

$$Kp = 7.2 \quad Ki = 36 \quad Kd = 0.36 \quad (53) \text{ Lazo interno Ball and Beam (Servomotor)}$$

La simulación y respuesta del controlador PID por el método de ziegler y nichols para el lazo externo del ball and se encuentra en las Figuras 57 y 58 (línea sólida referencia, línea punteada salida controlador) respectivamente y para el lazo interno del ball and beam (servomotor) en las Figuras 59 y 60 (línea sólida referencia, línea punteada salida controlador) respectivamente.

Figura 57. Diagrama de bloques PID método ziegler y nichols lazo externo Ball and Beam

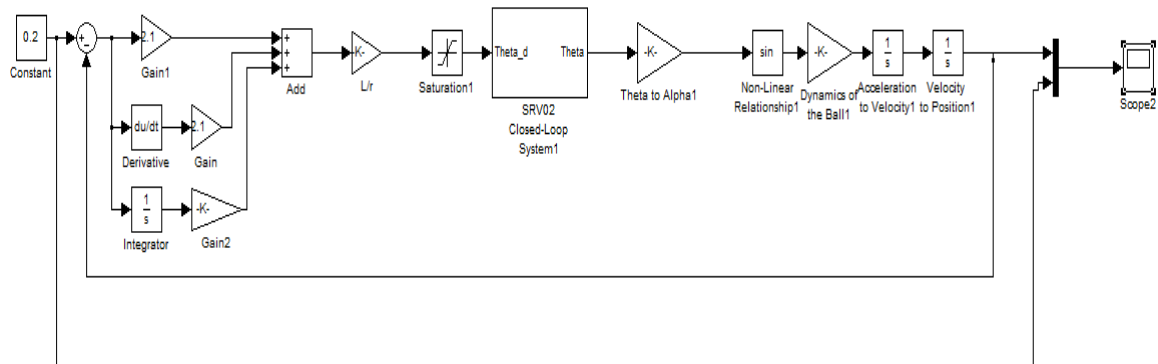


Figura 58. Respuesta método ziegler y nichols lazo externo Ball and Beam

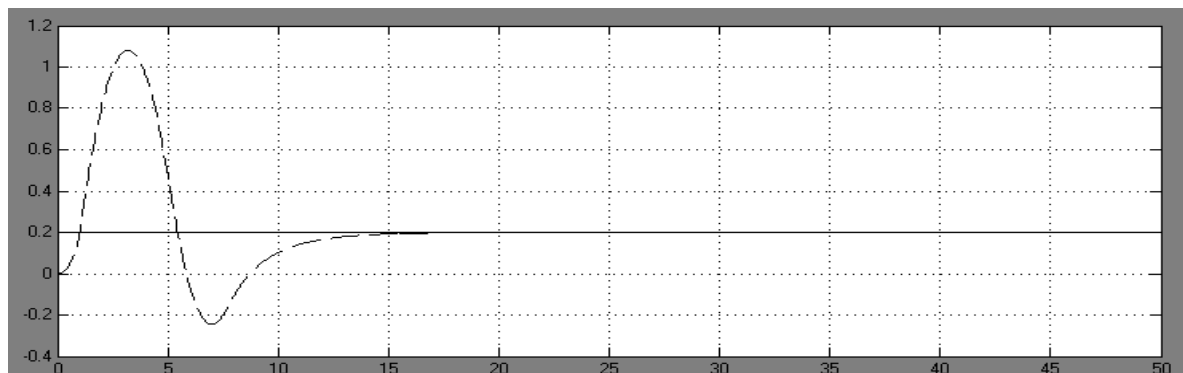


Figura 59. Diagrama de bloques PID método ziegler y nichols lazo interno Ball and Beam (Servomotor)

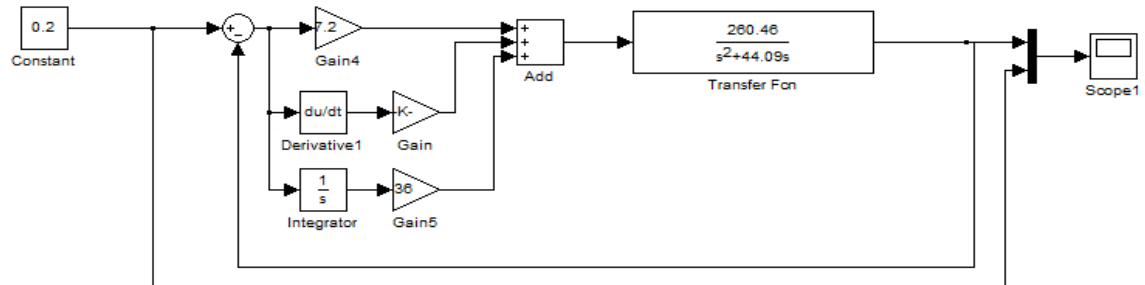
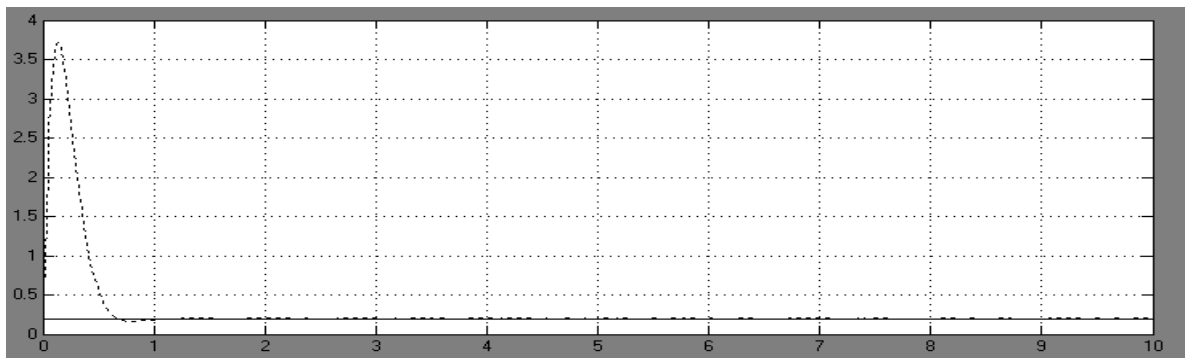


Figura 60. Respuesta método ziegler y nichols lazo interno Ball and Beam (Servomotor)



Tomando como punto de partida las ganancias obtenidas con ziegler y nichols se ajustan las ganancias para obtener una respuesta lo más satisfactoria posible del sistema controlado.

Se obtuvieron los siguientes resultados (Ecuación 54 y 55), al reemplazarlos en la ecuación 26:

$$K_p = 0.8 \quad k_i = 0.2 \quad k_d = 0.4 \quad (54) \quad \text{Lazo externo Ball and Beam}$$

$$K_p = 4.4 \quad k_i = 0.7 \quad k_d = 0.28 \quad (55) \quad \text{Lazo interno Ball and Beam (Servomotor)}$$

La simulación y respuesta del controlador PID por el ajuste de ganancias para el lazo externo del ball and beam se encuentra en las Figuras 61 y 62 (línea solida

referencia, línea punteada salida controlador) respectivamente y para el lazo interno del ball and beam (servomotor) en las Figuras 63 y 64 (línea sólida referencia, línea punteada salida controlador) respectivamente.

Figura 61. Diagrama de bloques PID método prueba y error lazo externo Ball and Beam

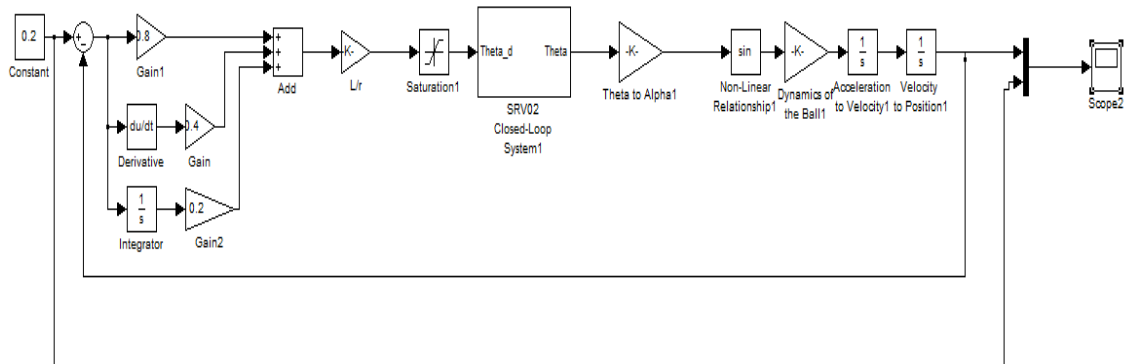


Figura 62. Respuesta método prueba y error lazo externo Ball and Beam

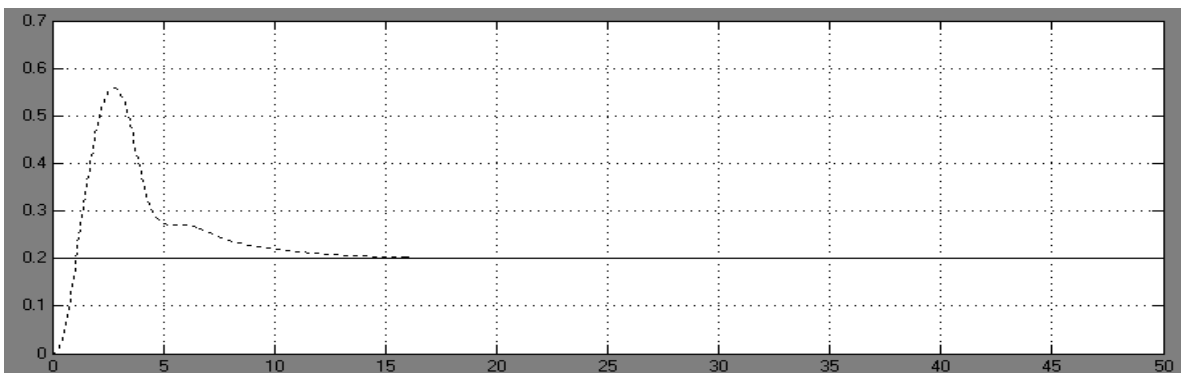


Figura 63. Diagrama de bloques PID ajuste ganancias lazo interno Ball and Beam (Servomotor)

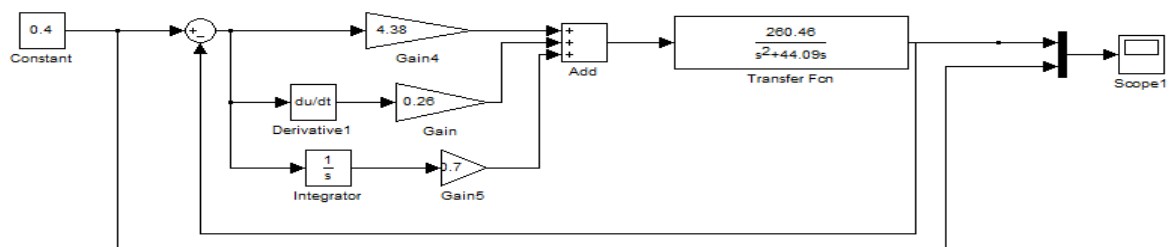
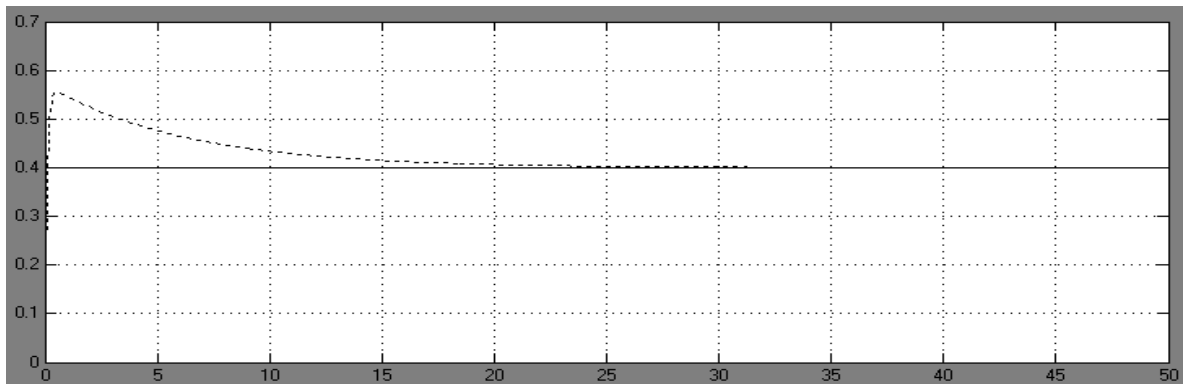


Figura 64. Respuesta ajuste ganancias lazo interno Ball and Beam (Servomotor)



En el Cuadro 7 y Cuadro 8 se encuentran consignados los valores obtenidos por los tres métodos para el lazo interno del Ball and Beam y para el lazo externo del Ball and Beam (Servomotor) respectivamente.

Cuadro 7. Valores PID lazo externo Ball and Beam

MÉTODO	Kp	Ki	Kd
Analítico	0.02		0.14
Ziegler y Nichols	2.1	0.525	2.1
Ajuste en línea	0.8	0.2	0.4

Cuadro 8. Valores PID lazo interno Ball and Beam (Servomotor)

MÉTODO	Kp	Ki	Kd
Analítico	0.00165		-0.16
Ziegler y Nichols	7.2	36	0.36
Ajuste en línea	4.4	0.7	0.28

7. IMPLEMENTACIÓN CONTROLADORES

En este apartado se explica el procedimiento y los elementos requeridos para realizar la implementación de los controladores diseñados, se hace mención del hardware y del software que originalmente traen los equipos de quanser y se describe con detalle cómo reemplazar estos equipos por los elementos que dispone la universidad autónoma para poder lograr que los módulos flexible Link, flexible Joint y ball and beam funcionen bajo otra plataforma sin cambiar su comportamiento natural.

7.1. HARDWARE DE QUANSER

Cada planta de quanser requiere un amplificador de potencia para poder funcionar. En este caso el fabricante ofrece una gran variedad de módulos de potencia que se pueden utilizar para conducir los motores. El modelo utilizado para nuestra implementación es el UPM-15-03 (Figura 65), este modelo provisiona la energía necesaria al motor SRV-02 y a los sensores analógicos y digitales de los módulos.

Figura 65. UPM-15-03



Fuente: UPM-15-03 [en línea] UPM power modules. Estados Unidos. [Consultado el 10 de enero de 2013]
http://quanser.com/english/html/solutions/power_units_UPM.asp

Para la adquisición de datos provenientes de los sensores analógicos de cada una de las plantas el UPM-15-03 posee entradas analógicas (S1, S2, S3, S4), que van conectadas directamente a los terminales de los sensores de cada módulo por medio de unos conectores tipo 6-pin-mini-DIN a 6-pin-mini-DIN (Figura 66).

Figura 66. Conectores tipo 6 pin mini DIN a 6 pin mini DIN



Posteriormente estas señales analógicas son acopladas a la tarjeta de adquisición de datos Quanser Q8 Terminal Board (Figura 67) por medio de un conector tipo 5-pin-DIN to 4xRCA (figura 68), esta tarjeta es la encargada de enlazar el modulo con el software de control QUARC que es el entorno de trabajo en matlab para trabajar cualquier tipo de experimento en los módulos flexible link, flexible joint y ball and beam.

Figura 67. Quanser Q8 terminal board



Fuente: Quanser Q8 terminal board [en línea] Data Acquisition system. Installing the data acquisition card [Consultado el 10 de enero de 2013] http://www.clemson.edu/ces/crb/ece495/References/manuals/quanser_q8_manual.pdf

Figura 68. Conector tipo 5-pin-DIN to 4xRCA



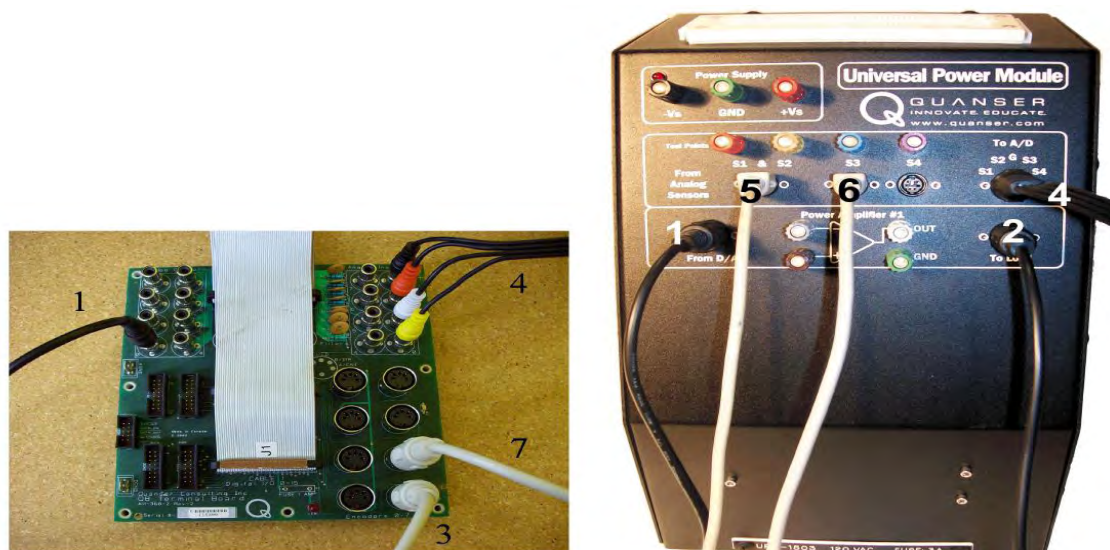
Para la adquisición de datos provenientes de los sensores digitales de cada una de las plantas las señales no pasan por el UPM-15-03 ya que la tarjeta de adquisición de datos Quanser Q8 Terminal Board se acopla a los sensores por medio de un conector 5-pin-stereo-DIN to 5-pin-stereo-DIN (figura 69), por medio de este conector se alimentan los encoders y se adquieren las señales de los canales A y B de dichos sensores.

Figura 69. Conector 5-pin-stereo-DIN to 5-pin-stereo-DIN



A continuación se presenta un esquema representativo sobre la forma en que se deben conectar los diferentes tipos de señales a la UPM-03 y a la Quanser Q8 terminal Board (Figura 70).

Figura 70. Esquema de conexión de las señales de los módulos



7.2. ADECUACIÓN PARA TRABAJAR CON LABVIEW

Para la implementación del experimento se hizo necesario reemplazar la forma de adquirir los datos de las plantas debido a que el uso del software QUARC genera un costo de licencia para la universidad, por lo que se optó por utilizar una tarjeta de adquisición de datos de la National Instrument que ya posee nuestro laboratorio y que puede ser utilizada a través del software LabView, a continuación se da una descripción de sus principales características técnicas (Cuadro 9) y se explica cómo se acoplo el sistema de quanser a la tarjeta National Instrument NI 6221.

Cuadro 9. Especificaciones técnicas de la tarjeta de adquisición de datos NI 6221.

Number of channels NI6220/6221 NI6224/6229 NI6225	8 differential or 16 single ended 16 differential or 32 single ended 40 differential or 80 single ended
ADC resolution	16 bits
DNI	No missing codes guaranteed
INI	Refer to the AI Absolute Accuracy Table
Sampling rate Maximum	250kS/s single channel, 250kS/s multi-channel (aggregate)

Minimum	No minimum
Timing accuracy	50 ppm of sample rate
Timing resolution	50ns
Input coupling	DC
Maximum working voltage for analog inputs	+11 V of AI GND
CMRR (DC to 60 Hz)	92 dB
Input impedance Device on AI+ to AI GND AI- to AI GND Device off AI+ to AI GND AI- to AI GND	>10 G Ω in parallel with 100pF >10 G Ω in parallel with 100pF 820 Ω 820 Ω
Input bias current	+100pA
Crosstalk (at 100 kHz) Adjacent channels Non-adjacent channels	-75dB -90dB
Small signal bandwidth (-3 dB)	700 KHz
Input FIFO size	4,095 samples
Scan list memory	4,095 entries
Data transfers PCI/PXI devices USB devices	DMA (scatter-gather, interrupts, programmed I/O) USB Signal Stream, programmed I/O

Acople y adquisición de datos de los módulos flexible link, flexible joint y ball and beam por medio de la tarjeta de adquisición de datos National Instrument NI 6221.

Para la adquisición de las señales analógicas de los módulos se utilizó el canal de entrada analógico 1 de la tarjeta, ubicada en el pin 33.

Para la adquisición de las señales de los encoders digitales de los módulos se utilizaron dos entradas digitales de la tarjeta, llamadas count 0 (pin 37 y pin 45) y count 1 (pin 42 y pin 46), estos canales son especialmente diseñados para realizar el conteo de pulsos, a continuación se presenta la forma como se adquieren las señales en la tarjeta NI 6221 (figura 71).

Para los encoders se hizo necesario anexar dos cables, 1 de alimentación de +5v y 1 de GND, los cuales provienen de los pines 14 y 13 respectivamente, estos cuatro cables (count 0, 5v y GND), fueron adaptados en un conector hembra de tipo 5-pin-stereo-DIN como se muestra en la figura 72

Figura 71. Adquisición señales Tarjeta NI 6221

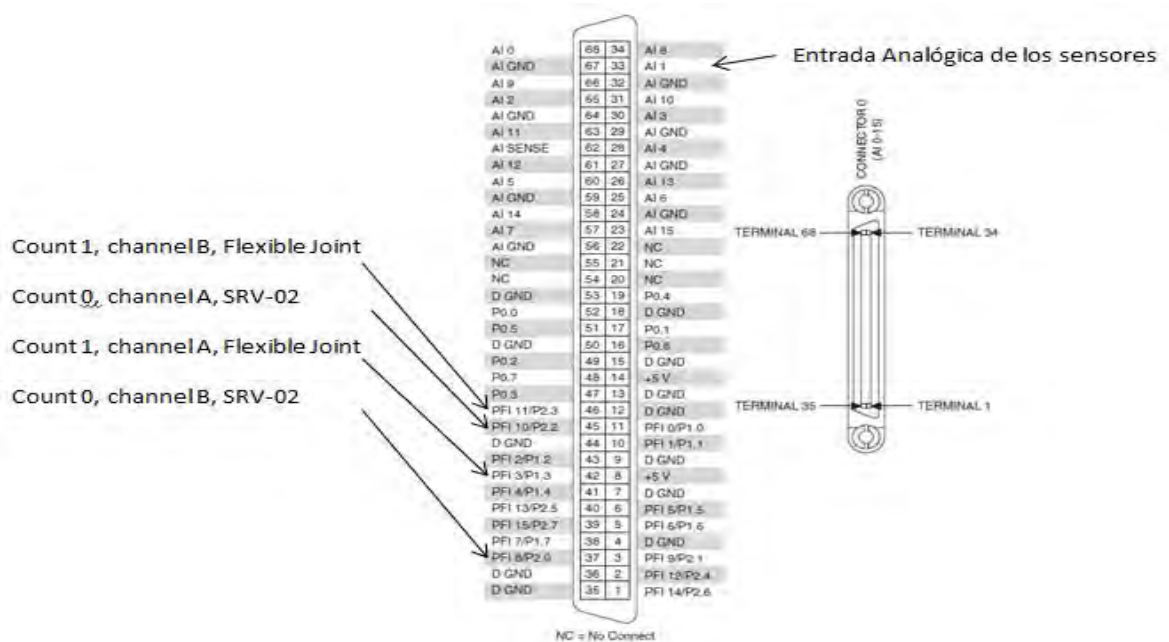
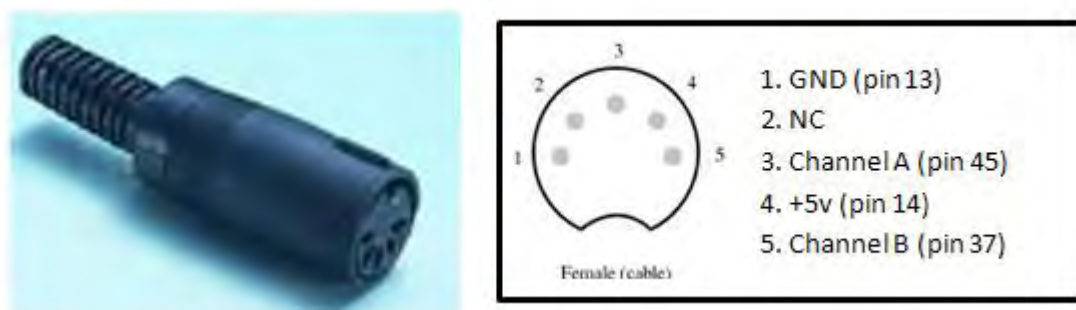


Figura 72. Configuración de la conexión para adquisición de las señales de encoders en NI 6221



Este procedimiento se repitió para configurar el count 1, canal por el cual se adquiere la señal del módulo Flexible Joint.

7.3. IMPLEMENTACIÓN CONTROLADOR DIFUSO

7.3.1. Lectura del ángulo del Flexible Link y Flexible Joint desde labview.

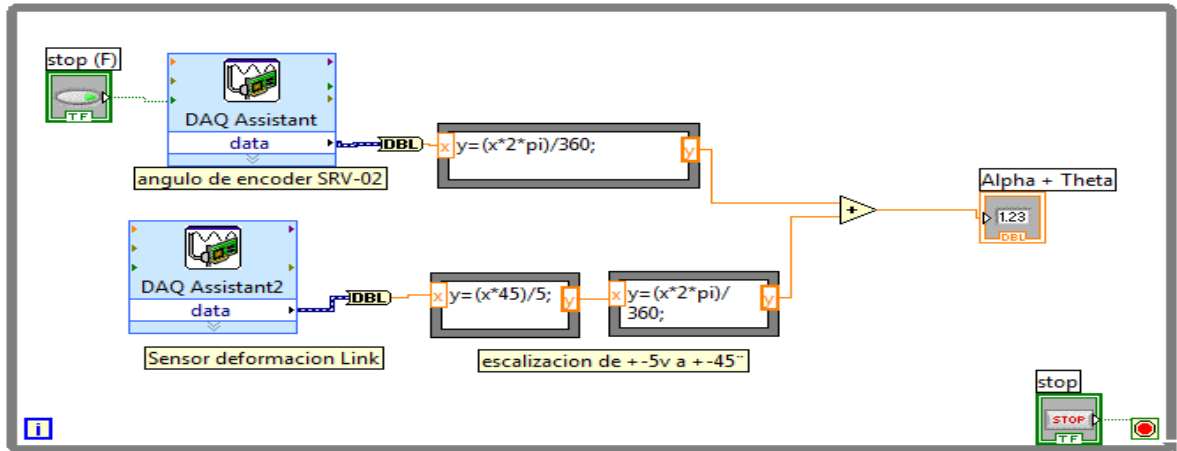
Partiendo de que para ambos módulos se pretende controlar su posición, la implementación para estos tiene iguales características desde el modelado, lo que hace la diferencia entre ambas implementaciones es la forma como debemos obtener las variable que se presentan en la función de transferencia de cada uno de ellos, es claro que en ambos casos la posición total del módulo depende de la sumatoria de dos ángulos como se describe en las ecuaciones 1 y 2. La salida total de la planta es la sumatoria del ángulo θ (posición del SRV-02)+ el ángulo α (deflexión del brazo), y para cada módulo la adquisición del ángulo α varia. A continuación se muestra la diferencia en la adquisición de esta variable para cada módulo.

MODULO FLEXIBLE LINK

El ángulo α que nos indica la deflexión del brazo de este módulo es una señal analógica entre el rango de +5v y -5v dependiendo de la dirección de la deflexión, para sumar este ángulo con el ángulo θ del SRV-02 que ya está presente en el count 0 de la tarjeta de adquisición de datos NI 6221 se recurrió a la ayuda del canal analógico 1 de la misma tarjeta y posteriormente se realizó la sumatoria de estas dos señales dentro de un ciclo infinito que garantiza la adquisición durante todo el tiempo de ejecución del experimento, en la figura 73 se ilustra la implementación de la sumatoria de los ángulos θ y α para el modulo flexible Link.

Debido a que la señal proveniente del brazo es una señal analógica de voltaje fue necesario escalizar esta señal para tener paridad de datos a la hora de realizar la sumatoria de los ángulos, para esto se creó un nodo de fórmula que convierte la señal de +5v a $+45^\circ$, lo que nos permite que las señales del motor y del brazo se puedan sumar sin generar errores de medidas.

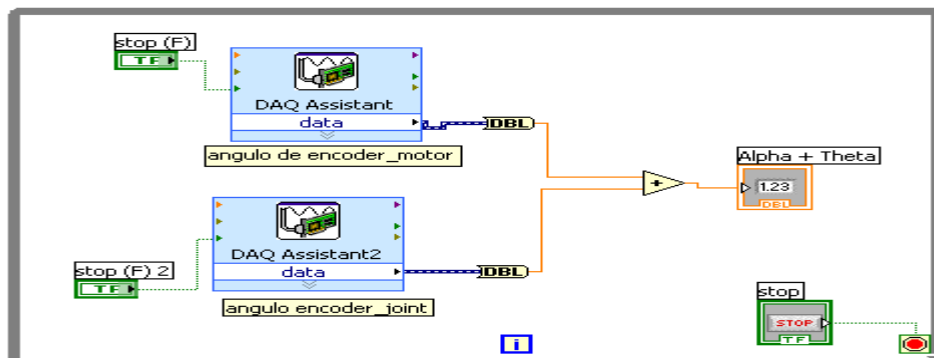
Figura 73. Sumatoria de los ángulos θ y α para el modulo flexible Link



MÓDULO FLEXIBLE JOINT

A diferencia del módulo anterior, el flexible Joint genera una señal digital proveniente de un encoder, por lo cual se facilita la sumatoria de el ángulo α y el ángulo θ del SRV-02 que ya está presente en el count 0 de la tarjeta de adquisición de datos NI 6221, en este caso basta solo con agregar la señal del count 1y realizar la sumatoria de estas dos señales dentro de un ciclo infinito que garantiza la adquisición durante todo el tiempo de ejecución del experimento, en la figura 74 se ilustra la implementación de la sumatoria de los ángulos θ y α para el modulo flexible Joint.

Figura 74. Sumatoria de los ángulos θ y α para el modulo flexible Joint



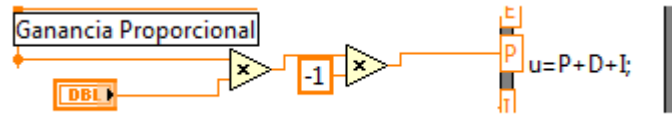
7.3.2. Implementación de controladores PID

7.3.2.1. Implementación controladores PID para los módulos Flexible Link, Flexible Joint y Ball and Beam. Se implementara un controlador PID en paralelo para los tres módulos que tiene la forma de la Ecuación 16.

La acción proporcional (Ecuación 56) es directamente proporcional a la señal de error por una constante que se denomina la ganancia proporcional, además se multiplica por -1 debido a que la planta tiene un comportamiento inverso. En la Figura 75 se observa la implementación de esta.

$$s = Kp * e \quad (56)$$

Figura 75. Implementación acción proporcional



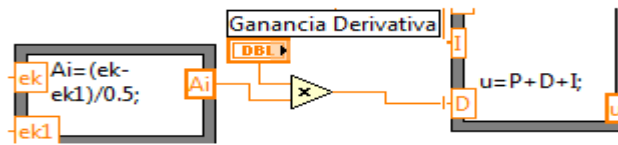
La acción derivativa (Ecuación 57) su salida es proporcional a la entrada, es decir es proporcional a la razón de cambio de la señal de error. Donde se multiplica por una constante denominada ganancia derivativa. En la Figura 76 se muestra la implementación de la ganancia derivativa.

$$s = Kd * \frac{de}{dt} \quad (57)$$

Dónde:

$$\frac{de}{dt} = \frac{e(k) - e(k - 1)}{T} \quad T: \text{Tiempo de muestreo}$$

Figura 76. Implementación acción derivativa



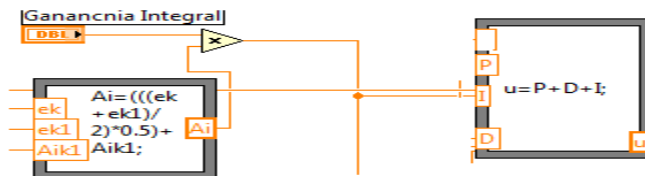
La acción integral (Ecuación 58) su salida es proporcional a la integral de la entrada en el tiempo. Donde se multiplica por una constante denominada ganancia integral. En la Figura 77 se observa la implementación de esta.

$$s = Ki * \int_0^t edt \quad (58)$$

Donde

$$\int e(k) = i(k-1) + \frac{T}{2}(e(k) + e(k-1)) \quad i(k-1): \text{Integral anterior}$$

Figura 77. Implementación acción integral



La suma de estas tres acciones da como resultado la acción de control del controlador PID. Cada una de las ganancias varía según el módulo (Flexible Link, Flexible Joint, Ball and Beam), que se esté controlando. Las respuestas y acciones de control de los controladores implementados se encuentran en las Figuras 78, 79 y 80 (línea solida referencia, línea punteada salida controlador, imagen de arriba respuesta e imagen de abajo acción de control).

Figura 78. Respuesta y acción de control controlador PID Flexible Joint

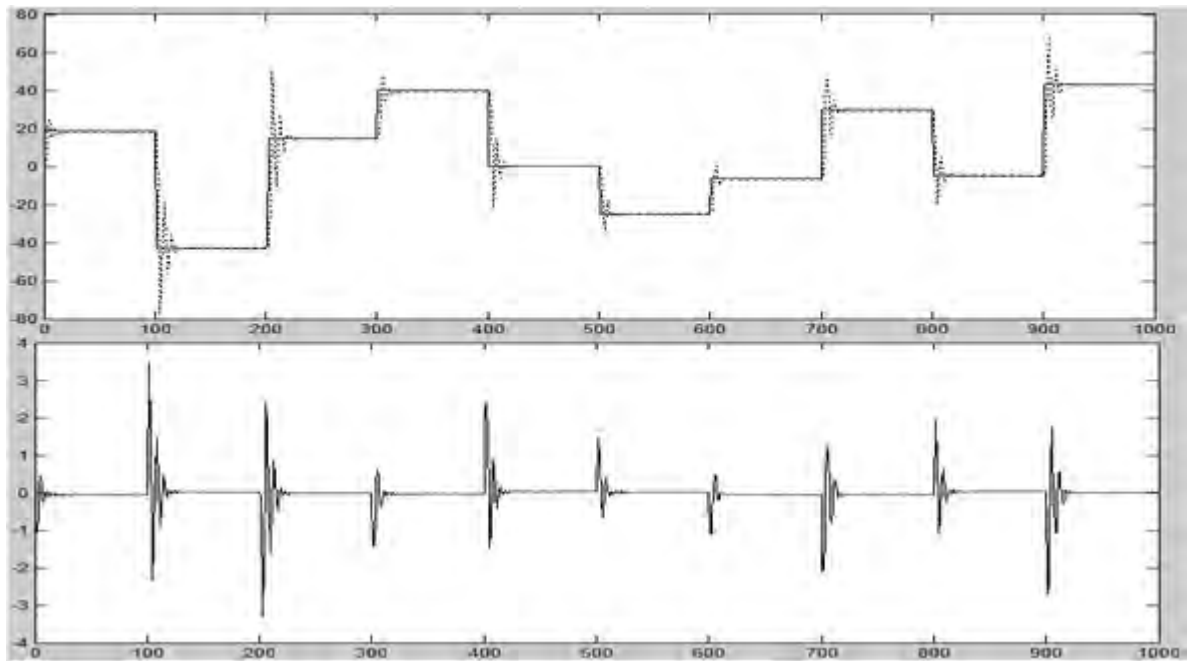


Figura 79. Respuesta y acción de control controlador PID Flexible Link

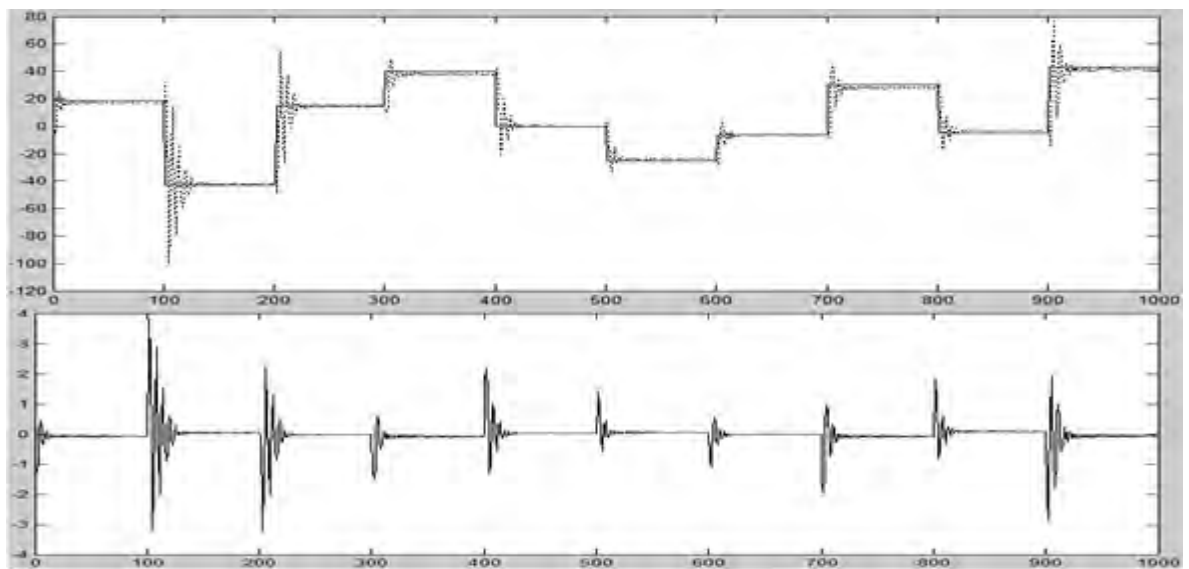
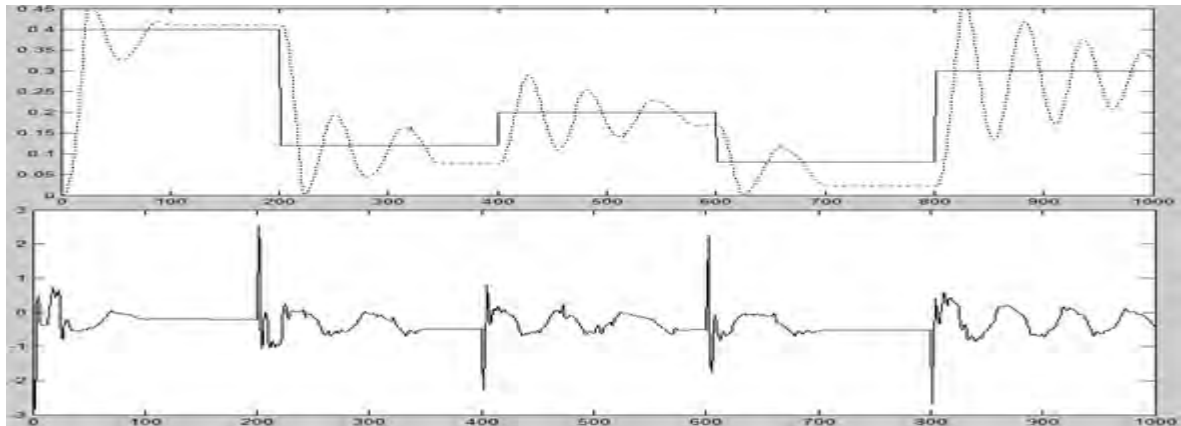


Figura 80. Respuesta y acción de control controlador PID Ball and Beam



7.3.3. Implementación de controladores difusos

7.3.3.1. Implementación del controlador difuso para los módulos Flexible Link y Flexible Joint. Para realizar la implementación de los controladores difusos de los módulos flexible link y flexible Joint fue necesario definir los conjuntos difusos que corresponden al error y a la derivada del error respectivamente ya que estas variables son necesarias para la realización de las reglas y del sistema de inferencia difuso en el software Labview.

Con la ayuda del software Matlab se establecieron los parámetros de diseño que representan los conjuntos difusos del controlador, las funciones trapezoidales y triangulares con cada una de las etiquetas lingüísticas fueron generadas y graficadas de acuerdo a los siguientes parámetros y condiciones. Posteriormente estos conjuntos difusos se implementaron en Labview

Para el control de posición del módulo se determinó una zona de trabajo comprendida entre 45° y -45° , para el diseño, las unidades manejadas son radianes por lo cual la zona de trabajo del módulo está comprendida entre $\pi/4$ y $-\pi/4$.

Se trabajó con una derivada del error normalizada entre 1 y -1 con igual cantidad de conjuntos difusos que el error (5 conjuntos).

Después de determinar los conjuntos difusos con funciones trapezoidales y triangulares, se pasó a una representación con campanas gaussianas ya que esto facilitó la implementación en labview, a continuación se muestra cómo se generan

las funciones que representan cada uno de los trapezoides, triángulos, campanas gaussianas y gaussianas en el Software Matlab.

Función trapezoidal = `trapmf(x,[-pi/4 -pi/4 -pi/6 -pi/12]);`

Función de Campana de gauss = `gbellmf(x,[-pi/12 3.5 (-5*pi/24)]);`

Función triangular = `trimf(x,[-pi/6 -pi/12 0]);`

Función de gauss = `gaussmf(x,[-pi/26 -pi/12]);`

En el Anexo C se muestra el código que permite generar cada uno de los conjuntos difusos si se utilizan las funciones mencionadas con anterioridad en la figura 81 y 82 se muestra como cada función de campana de gauss y cada función de gauss representa a los conjuntos originales de las funciones trapezoidales y triangulares respectivamente.

Figura 81. Campanas gaussianas error Flexible Link y Flexible Joint

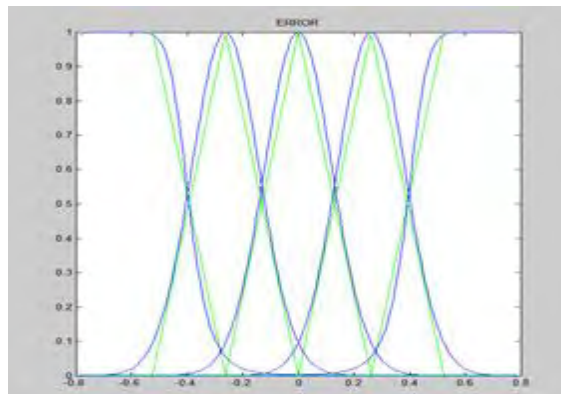
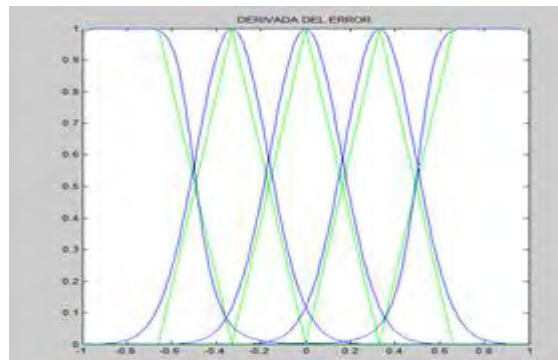


Figura 82. Campanas gaussianas derivada del error



Después de realizar la representación gráfica de los conjuntos difusos se pasó a la implementación de los mismos en el software Labview, para ello se constituyó un nodo de fórmula que contiene cada uno de los conjuntos difusos representados por las siguientes ecuaciones.

La función de las campanas de gauss (Ecuación 59) recibe los parámetros que previamente fueron calculados en el código de matlab, estos son:

Ancho = α
 Caída= b
 Centro = c

X es la variable a evaluar en la función, para nuestro caso es el error o la derivada del error respectivamente para cada nodo de formula.

$$f(x, a, b, c) = \frac{1}{1 + \left| \frac{x - c}{a} \right|^{2b}} \quad (59)$$

La función de gauss (Ecuación 60) representa los conjuntos difusos gaussianos calculados con anterioridad en el código de matlab y está definido por los siguientes parámetros y variables.

Ancho = a
 Centro = σ

X es la variable independiente que se evalúa en la función, dependiendo del nodo de formula esta variable es el error o la derivada del error.

$$f(x, \delta, c) = e^{\frac{-(x-c)^2}{2\delta^2}} \quad (60)$$

Cada una de estas ecuaciones se ve representada en los nodos de fórmula de labview de la siguiente manera.

$$f(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \longrightarrow s1 = 1/(1+abs((x-(c))/ \alpha)^{2*b}) ;$$

$$f(x; \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}} \longrightarrow s2 = \exp(-0.5*((x-(c))/ \sigma)^{2});$$

S1 y s2 son las ecuaciones que representan las funciones de campanas de gauss y las funciones de gauss respectivamente para el nodo de fórmula del error y el nodo de fórmula de la derivada de error, a continuación en la figura 83 se muestra el nodo de fórmula correspondiente a el error y en la figura 84 el nodo de formula correspondiente a la derivada del error.

Figura 83. Nodo de fórmula que representa los conjuntos difusos del error

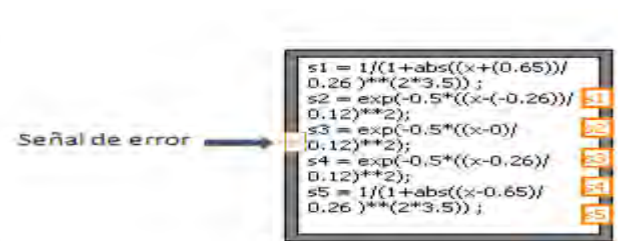
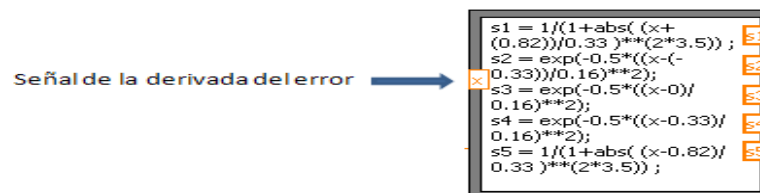


Figura 84. Nodo de fórmula que representa los conjuntos difusos de la derivada del error



Las respuestas y acciones de control de los controladores difusos implementados para el Flexible Link y Flexible Joint se pueden observar en las Figuras 85 y 86 (línea solida referencia, línea punteada salida controlador, imagen de arriba respuesta e imagen de abajo acción de control) respectivamente.

Figura 85. Respuesta y acción de control controlador difuso Flexible Link

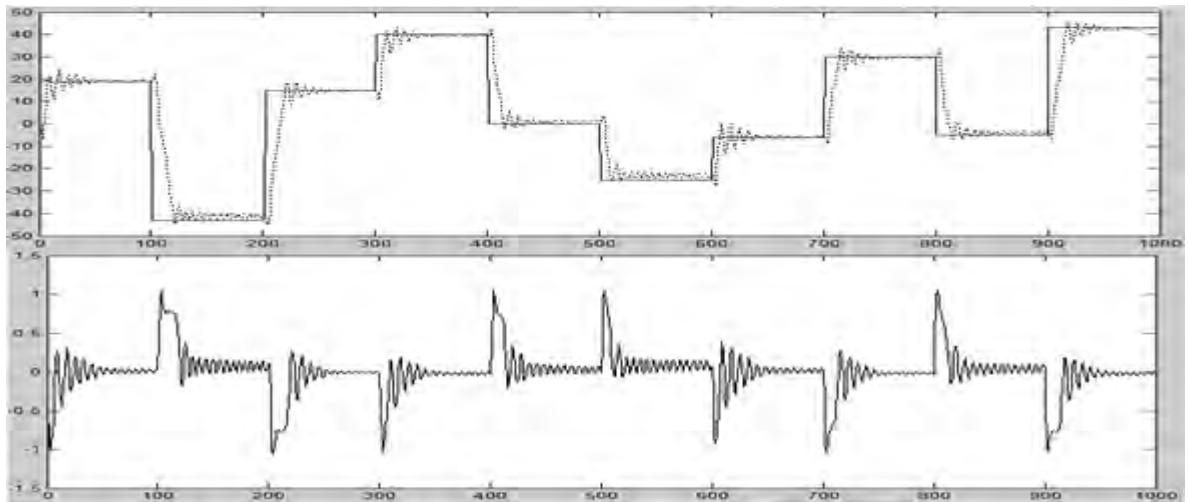
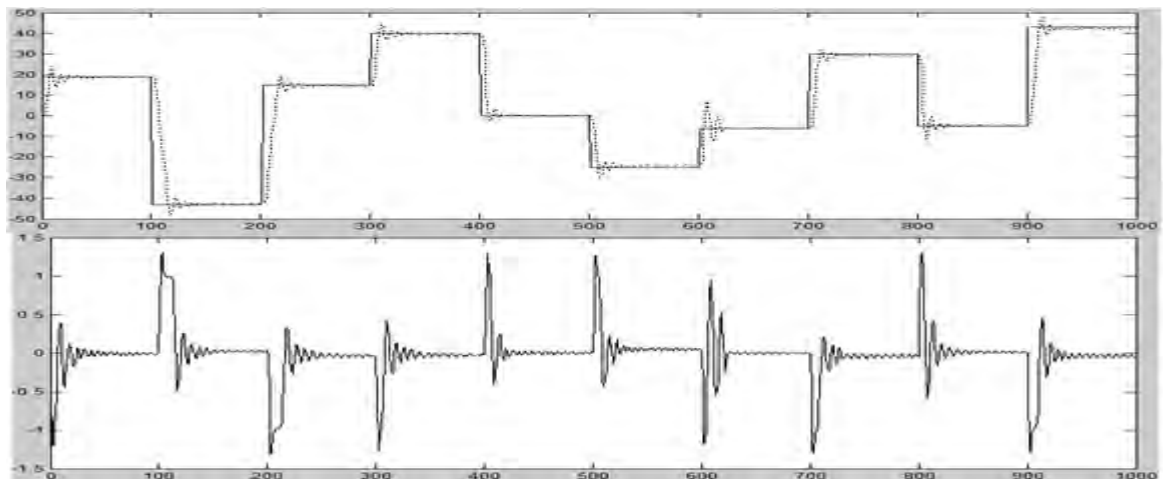


Figura 86. Respuesta y acción de control controlador difuso Flexible Joint

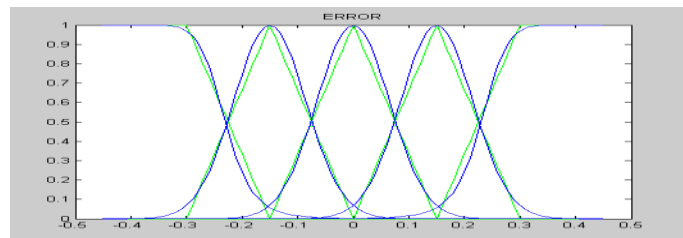


7.3.4. Implementación del controlador difuso para el módulo Ball and Beam.

Para el módulo ball and beam se realizó la implementación del controlador partiendo de la generación de los conjuntos difusos de la misma manera como ya se describió en el módulo flexible link, esta implementación difiere de la ya mencionada en que los parámetros de creación de los conjuntos difusos están regidos por las siguientes condiciones

Para el control de posición del lazo externo del sistema, o sea la posición de la bola sobre la viga, los conjuntos difusos se generaron en un rango de trabajo entre 0.45 m y -0.45 m como se muestran en la figura 87, esto equivale a los valores máximos que puede llegar a tomar el error en cualquier momento del experimento, en el Anexo D se muestra el código de matlab que permite generar la representación gráfica de estos conjuntos.

Figura 87. Campanas gaussianas error Ball and Beam



Se trabajó con una derivada del error normalizada entre 1 y -1 con igual cantidad de conjuntos difusos que el error (5 conjuntos), esto es posible observarlo en la Figura 79 ya que ambas implementaciones de los controladores se rigen bajo el mismo comportamiento de la derivada del error.

Al igual que en el módulo flexible link después de determinar los conjuntos difusos con funciones trapezoidales y triangulares, se pasó a una representación con campanas gaussianas, a continuación en las figuras 88 y 89 se muestra cómo quedó la representación de los conjuntos difusos en labview después de reemplazarlos con la ayuda de las ecuaciones 59 y 60 (ecuaciones de la campana de gauss y de la función de gauss).

Figura 88. Nodo de fórmula que representa los conjuntos difusos del error en el Ball and Beam

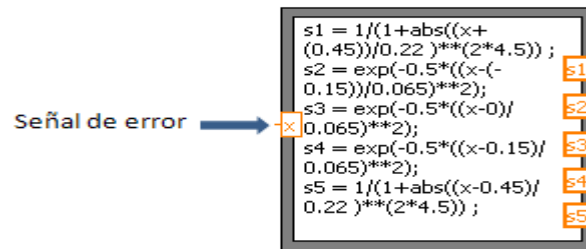
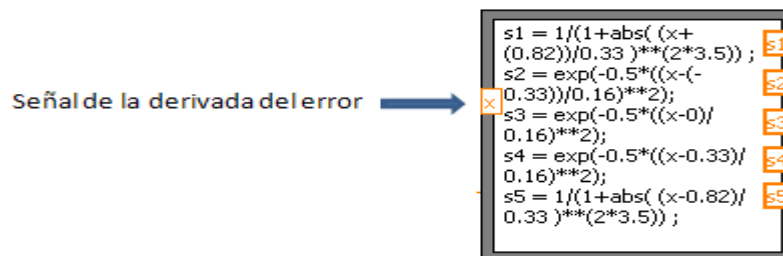
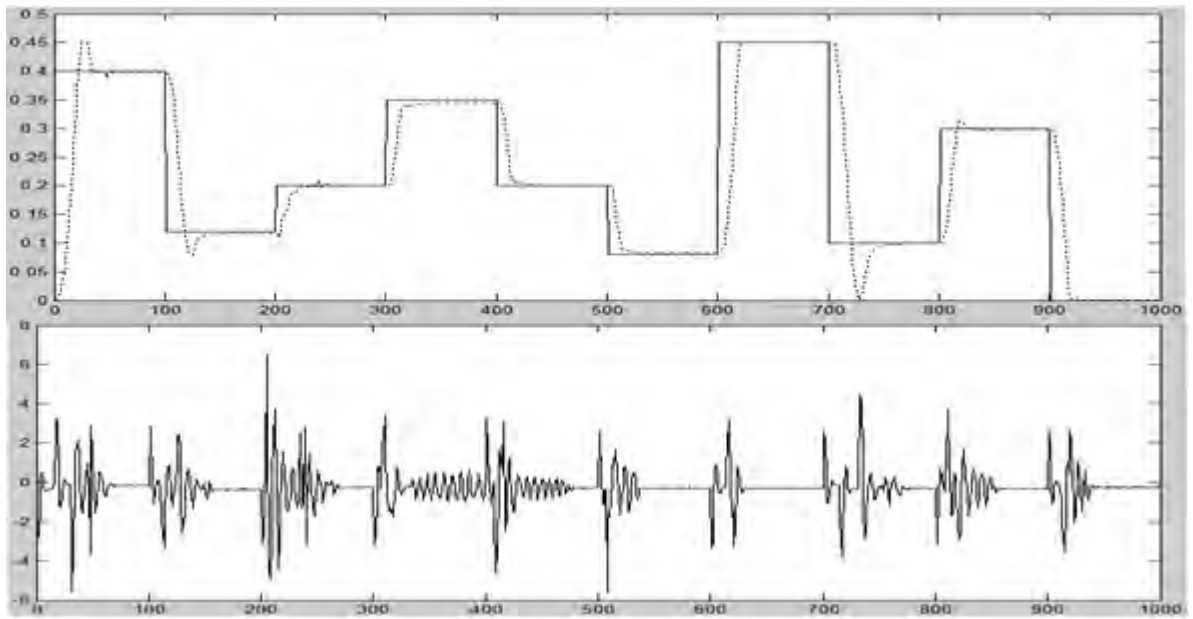


Figura 89. Nodo de fórmula que representa los conjuntos difusos de la derivada del error en el ball and beam



La respuesta del controlador difuso implementado se puede observar en la Figura 90 (línea sólida referencia, línea punteada salida controlador, imagen de arriba respuesta e imagen de abajo acción de control).

Figura 90. Respuesta y acción de control controlador difuso Ball and Beam



8. VIRTUALIZACION DE LOS MÓDULOS FLEXIBLE LINK Y FLEXIBLE JOINT

Para el diseño de la virtualización para los módulos Flexible Link y Flexible Joint, se implementara cada uno de los controladores diseñados anteriormente para estos, se realizará una animación, habrá gráficas para observar su comportamiento y una referencia para poder variar los valores y observar el control.

En el panel de control se podrá escoger el tipo de controlador que se desea, ya sea el controlador PID o el controlador difuso, estos estarán separados por medio de pestañas. Los parámetros de los controladores se podrán modificar según desee el usuario.

En la animación se encontrará la virtualización de los módulos Flexible Link y Flexible Joint, para mostrar el comportamiento de la planta cuando se efectué la acción de control.

En las gráficas se podrá observar la acción de control y el comportamiento de la planta de la variable controlada, esta mostrará la referencia y como el controlador sigue esta referencia cuando se activa.

La referencia contará con un deslizador para para modificar los valores deseados por el usuario, también habrá dos botones uno para parar o empezar la simulación y otro para resetear todo el proceso.

8.1. VIRTUALIZACIÓN DEL MÓDULO FLEXIBLE LINK

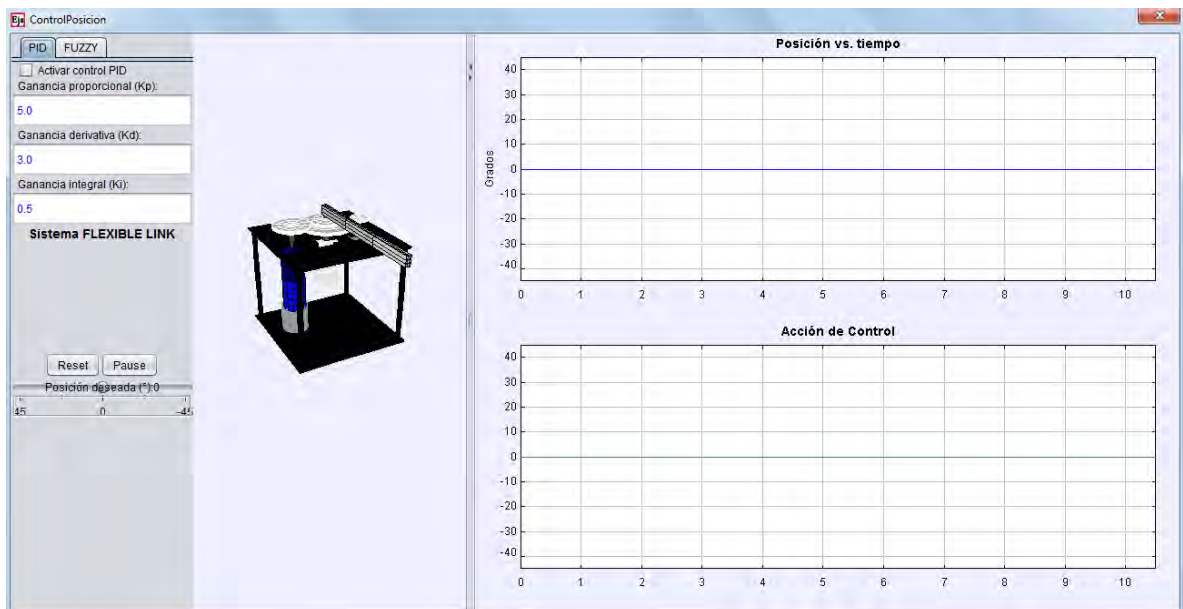
Esta contará con una ventana principal en donde se encontrará una foto del módulo, la función de transferencia que determina su comportamiento y un botón para poder determinar el tipo de control que se desea, en este caso será un control de posición. En la Figura 91 se muestra como es esta ventana.

Figura 91. Ventana principal Flexible Link



Al activar el control de posición se abrirá una segunda ventana, donde se encontrara la animación, panel de control, gráficas y referencia, como se muestra en la Figura 92.

Figura 92. Ventana secundaria Flexible Link.



A continuación se presentará detalladamente cada uno de estos elemento que hacen parte de esta ventana.

En la primera columna de la ventana se encuentran las opciones de qué tipo de control se le puede hacer al módulo, estos se encuentran separados por pestañas para que el usuario determine qué tipo de control se le va a realizar, en cada controlador se puede variar sus ganancias para observar cómo responde el módulo ante estas variaciones. En la Figura 93 se muestra el panel de control para ambos controladores.

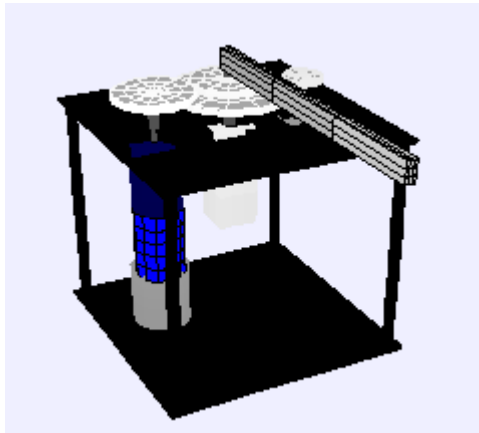
Figura 93. Panel Control

En esta misma columna se encuentra la referencia representada por un deslizador que permite ingresar la posición en grados deseada (Figura 94).

Figura 94. Referencia

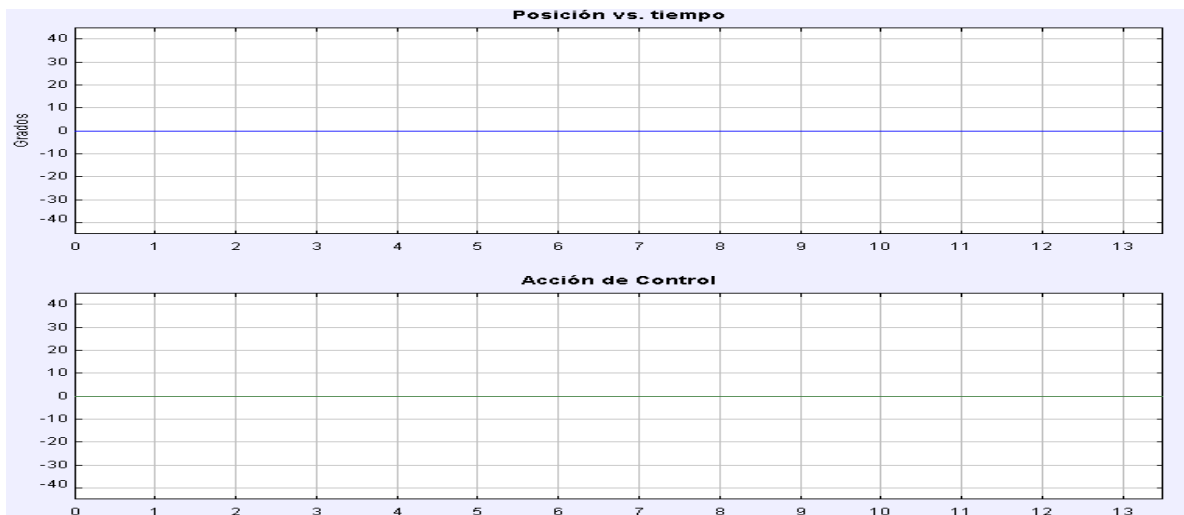
En la segunda columna de esta ventana, se encuentra la animación del módulo que se realizó por medio de elementos 3D, que muestra una representación similar a la planta física, para simular los grados que se mueve la barra y el usuario puede observar de forma gráfica el comportamiento de la planta al aplicar el control, en la Figura 95 se muestra la animación de este.

Figura 95. Animación Flexible Link



En la columna tres se observa las gráficas que corresponde al comportamiento de la variable controlada con respecto al tiempo y la acción de control que se realiza para llevar a la planta a la posición deseada. En la Figura 96 se muestra estas dos gráficas.

Figura 96. Gráficas.



8.2. VIRTUALIZACIÓN DEL MÓDULO FLEXIBLE JOINT

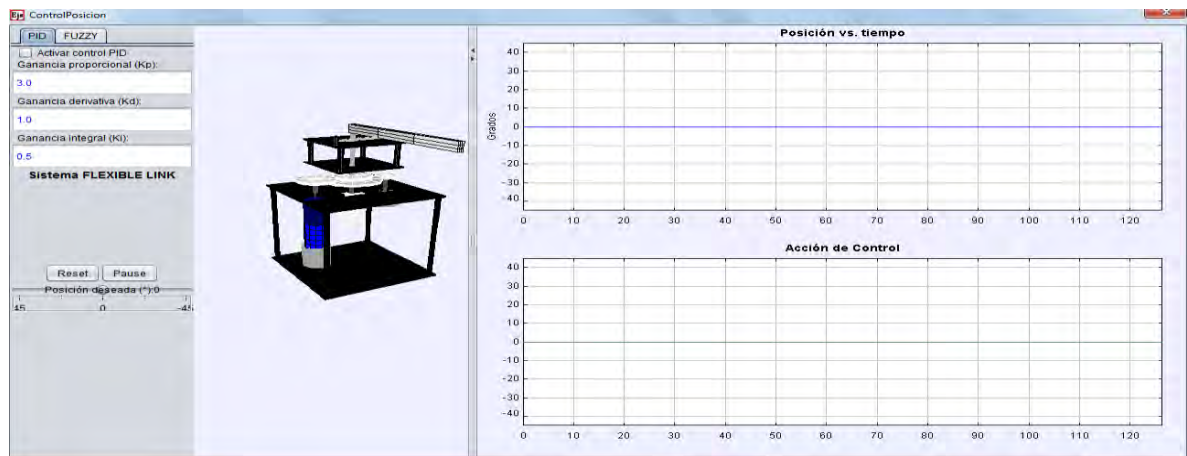
Al igual que el flexible link este contara con dos ventanas una principal para determinar el tipo de control que se va realizar en este caso solo será control de

posición (Figura 97) y una secundaria (Figura 98) donde se encontrara la animación, controladores, referencia y gráficas.

Figura 97. Ventana principal Flexible Joint

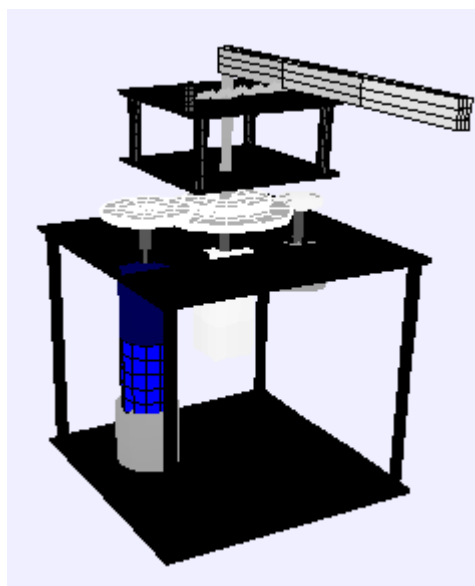


Figura 98. Ventana secundaria Flexible Joint



Esta interfaz es idéntica a la anterior solo cambia la animación del módulo, este se virtualiza como es en realidad el Flexible Joint, en la Figura 99 se muestra esta virtualización.

Figura 99. Animación del Flexible Joint.



9. ANÁLISIS COMPARATIVO

El análisis comparativo se realizara comparando el desempeño de cada uno de los controladores diseñados e implementados para cada uno de los módulos con los índices de desempeño.

Los índices de desempeño se usan para comparar la calidad de regulación del controlador y el esfuerzo de control necesario para alcanzarla, los índices que se usaran serán: el índice de desempeño de error y el índice de desempeño de esfuerzo de control.

Índice de desempeño de error: Este se encarga de evaluar la diferencia entre el valor deseado y el valor obtenido en la salida. Se definen dos índices, el promedio del error al cuadrado (Ecuación 61) y el promedio del error absoluto (Ecuación 62).

$$I1 = \frac{1}{N} \sum_{K=0}^N (r(k) - y(k))^2 \quad (61)$$

$$I2 = \frac{1}{N} \sum_{K=0}^N |r(k) - y(k)| \quad (62)$$

Índice de desempeño de esfuerzo de control: Evalúa el promedio del valor absoluto de la acción de control (Ecuación 63)

$$I3 = \frac{1}{N} \sum_{K=0}^N |u(k)| \quad (63)$$

Índice suavidad esfuerzo de control: Compara la diferencia entre una muestra anterior de la señal de control y la actual (ecuación 64). Permite conocer que tan bruscos son los cambios de la señal de control.

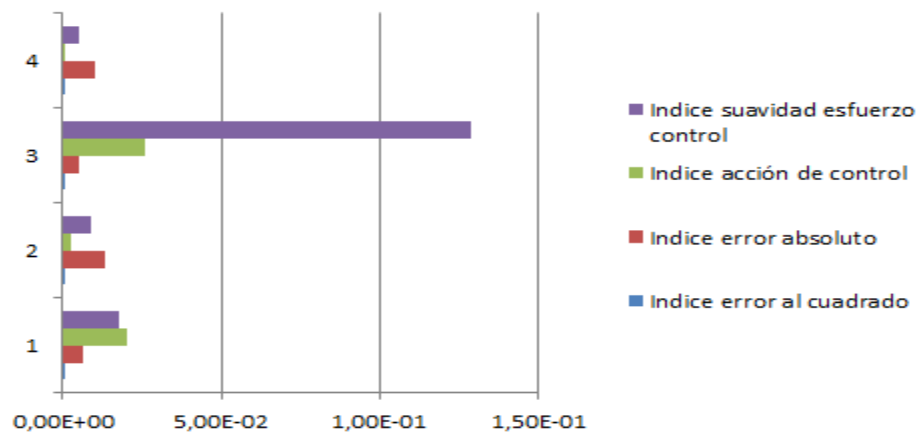
$$I4 = \frac{1}{N-1} \sum_{K=0}^N (u(k) - u(k-1))^2 \quad (64)$$

Para realización del experimento en condiciones iguales para cada uno de los controladores se generó un vector de valores randomicos con los cuales se pudo garantizar que cada módulo respondiera exactamente ante la misma entrada de setpoint. Se obtuvieron los siguientes resultados (Cuadro 10) para el Flexible Joint, (Cuadro 11) para el Flexible Link y (Cuadro 12) Ball and Beam.

Cuadro 10. Índices desempeño Flexible Joint

INDICE	CONTROLADOR DIFUSO Implementación (1)	CONTROLADOR DIUSO Simulación (2)	CONTROLADOR PID Implementación (3)	CONTROLADOR PID Simulación (4)
I1	4.53E-05	1.83E-04	2.79E-05	9.84E-05
I2	0.006730675	0.013558923	0.005285693	0.009919946
I3	0.020592511	0.002716784	0.026070442	0.000155839
I4	0.018105898	0.008911122	0.128949071	0.00503373

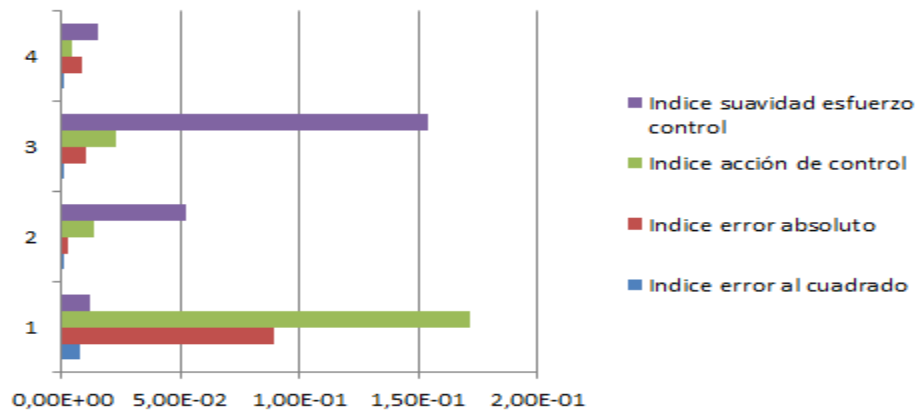
Figura 100. Gráfica Índices desempeños Flexible Joint



Cuadro 11. Índices desempeño Flexible Link

INDICE	CONTROLADOR DIFUSO Implementación (1)	CONTROLADOR DIUSO Simulación (2)	CONTROLADOR PID Implementación (3)	CONTROLADOR PID Simulación (4)
I1	7.93E-03	7.58E-06	1.007E-04	7.88E-05
I2	0.089070883	0.002753816	0.01003679	0.00887822
I3	0.004169712	0.013990981	0.022494919	0.004197569
I4	0.011668185	0.052245705	0.154417632	0.015138669

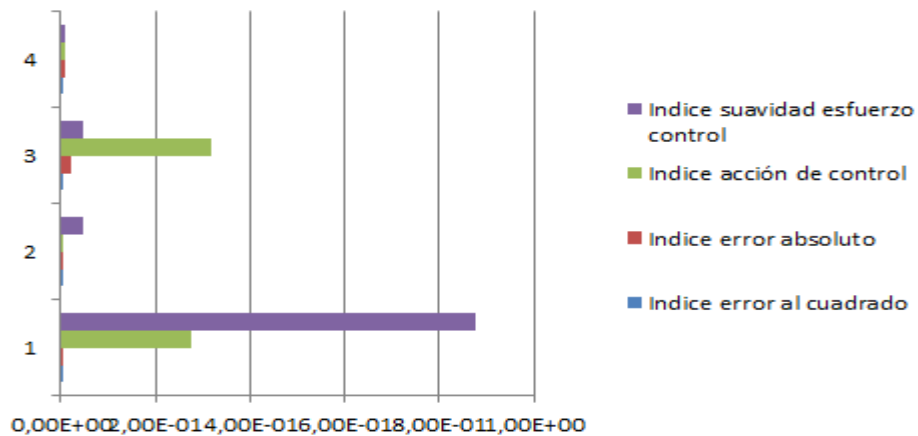
Figura 101. Gráficas Índices de desempeño Flexible Link



Cuadro 12. Índices desempeño Ball and Beam

INDIC E	CONTROLADO R DIFUSO Implementación (1)	CONTROLADO R DIUSO Simulación (2)	CONTROLADO R PID Implementación (3)	CONTROLADO R PID Simulación (4)
I1	2.83E-03	1.37E-06	3.96E-04	1.00E-04
I2	0.002815009	0.001173602	0.019921686	0.010012917
I3	0.275428646	5.45E-06	0.318358759	0.010318348
I4	0.875694283	0.049278118	0.049269941	0.007984772

Figura 102. Gráficas Índice de desempeño Ball and Beam



Como se observa en cada una de las gráficas 100, 101 y 102, se puede concluir que los controladores en la simulación tienen un mejor desempeño que los controladores implementados debido a que en todos los casos los índices de

desempeño son menores en los controladores en la simulación que los implementados, esto se debe a que los controladores en la simulación están en condiciones ideales y no se ve afectado por el entorno.

Al comparar los índices de desempeño del error se puede concluir que el controlador implementado PID tiene un mejor desempeño con los módulos Flexible Link y Flexible Joint debido a que el error es mucho menor en estos controladores, mientras que en el caso del módulo Ball and Beam tiene un mejor desempeño el controlador difuso. Los controladores simulados tienen el mismo comportamiento en los índices de desempeño del error al igual que los controladores implementados, ya que el PID tiene un mejor desempeño en el flexible link y flexible joint, y el difuso en el ball and beam. Al tener un mejor desempeño cada uno de los controladores según sea el caso mencionado anteriormente, significa que al tener un error más pequeño la salida del controlador es muy parecida a la referencia, logrando un mejor control.

Al comparar los índices de la acción de control en los controladores implementados, se puede concluir que el controlador difuso tiene un mejor desempeño que controlador PID en el módulo Flexible Joint y Ball and Beam, mientras que en el módulo Flexible Link la acción de control tiene un mejor desempeño en el controlador PID que en el controlador Difuso. El índice de la acción de control en los controladores en simulación es muy parecida a los implementados con la diferencia que este índice tiene un mejor desempeño en el módulo flexible joint con el PID, mientras en los otros dos casos es idéntico en los controladores implementados y en simulación, teniendo un mejor desempeño el controlador difuso en simulación con el ball and beam y flexible link con el PID. Debido a que gasta menos energía al realizar la acción de control según sea el caso. También se puede concluir que los controladores que tienen mejor desempeño en el índice del error, son los que generan más cambios bruscos en la señal de control, debido a que el índice de suavidad del esfuerzo de control es mayor en todos los casos.

10. CONCLUSIONES

Con la implementación de los controladores diseñados en el software Labview se logró la independización de software Quarc y de la tarjeta de adquisición de datos Quanser Q8, logrando a través de ello desvincular el uso de los módulos de las licencias del fabricante.

El uso de diferentes de módulos permitió un mejor análisis de los comportamientos de los controladores clásicos e inteligentes, debido a que la dinámica de cada módulo varía y presenta diferentes retos para cada técnica de control.

La realización de los módulos virtuales sirvió de apoyo para vislumbrar el comportamiento de cada uno de los controladores diseñados antes de su implementación real. Los módulos virtuales son una herramienta didáctica de gran ayuda para los casos en los que no se tiene acceso a los equipos que se pretenden utilizar. La virtualización permitirá la interacción del usuario con cada uno de los módulos.

Realizar un experimento bajo condiciones iguales para cada módulo y para cada tipo de controlador, permitió determinar cuál de ellos funciona mejor de acuerdo a la dinámica de cada módulo.

BIBLIOGRAFIA.

- [1] BOLAÑOS V William E, "Diseño e implementación de un Laboratorio Remoto de Control Inteligente", Cali, 2009. Trabajo de grado (Ingeniero Mecatrónico). Universidad Autónoma de Occidente. Facultad de Ingeniería. Departamento de Automática y Electrónica.
- [2] COTERO O. José B, Capítulo 2 Control clásico/moderno y control inteligente [en línea]
<<http://www.mde.iteso.mx/titulacion/estudio%20de%20caso/ejemplos/marco%20teorico%20en%20progreso.pdf>> [citado el 20 de mayo de 2012]
- [3] (-----). Control clásico y control inteligente. [en línea]
<<http://www.mde.iteso.mx/titulacion/estudio%20de%20caso/ejemplos/marco%20teorico%20en%20progreso.pdf>> [citado el 19 de noviembre de 2012]
- [4] DELGADO G Mónica A, "Laboratorio Virtual de Control Inteligente", Cali, 2009. Trabajo de grado (Ingeniero Mecatrónico). Universidad Autónoma de Occidente. Facultad de Ingeniería. Departamento de Automática y Electrónica.
- [5] Departamento de informática y automática de la UNED. Laboratorio Virtual. [en línea] < <http://lab.dia.uned.es/automatlab/informacion.html> > [citado el 29 de mayo de 2012]
- [6] Easy Java Simulations [en línea] Disponible en Internet:
<<http://www.um.es/fem/EjsWiki/Main/WhatIsEJS?>>[Citado el 19 de mayo de 2012].
- [7] JUÁREZ H José M, CAÑADAS M José J, MARÍN M Roque, Capítulo 14 Control [en línea] <
http://novella.mhhe.com/sites/dl/free/8448156188/592196/8448156188_CapMuestra.pdf > [citado el 14 de noviembre de 2012]
- [8] LIBREROS José, "Laboratorio Virtual de Redes Neuronales", Cali, 2009. Trabajo de grado (Ingeniero Mecatrónico). Universidad Autónoma de Occidente. Facultad de Ingeniería. Departamento de Automática y Electrónica.

[9] MAZUERA Alexander, “Librería de Control Difuso para los PLC de siemens”, Cali, 2009. Trabajo de grado (Ingeniero Mecatrónico). Universidad Autónoma de Occidente. Facultad de Ingeniería. Departamento de Automática y Electrónica.

[10] QUANSER. Flexible Joint [en línea] <http://www.quanser.com/english/html/products/fs_product_challenge.asp?lang_code=english&pcat_code=exp-rot&prod_code=R4-flexint > [citado el 19 mayo de 2012].

[11] (-----). Flexible Link [en línea] <http://www.quanser.com/english/html/products/fs_product_challenge.asp?lang_code=english&pcat_code=exp-rot&prod_code=R3-flexlink > [citado el 19 de mayo de 2012]

[12] (-----). Servo-Planta SRV02 [en línea] <http://www.quanser.com/english/downloads/products/Rotary/SRV02_ROTIFLEX_PIS_031108.pdf > [citado el 19 de mayo de 2012].

[13] VALLEJO R. Erick, MÉNDEZ Fernando, OCHOA Carlos. “Aplicación del control difuso al diseño e implementación de un módulo de potencia”. Universidad del Norte. [en línea] <http://ciruelo.uninorte.edu.co/pdf/ingenieria_desarrollo/10/aplicacion_del_control_difusion.pdf > [citado el 19 de mayo de 2012].

[14] VIVAS A Oscar A, “Control predictivo de un robot tipo scara”. Revista chilena de ingeniería. 2006.

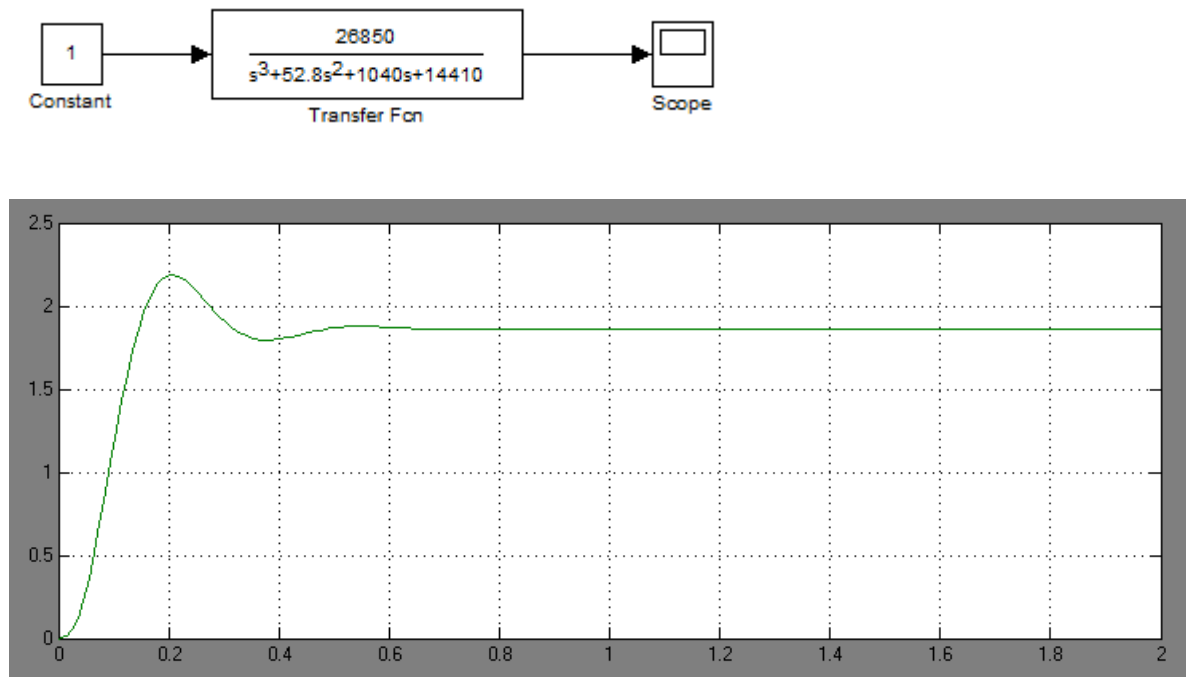
ANEXOS

Anexo A. Flexible Joint Segundo Orden

Para llevar la función de transferencia del flexible joint (Ecuación 8) que esta de cuarto orden a una de segundo orden, primero se saca el integrador puro que tiene, quedando de tercer orden y con la respuesta en lazo abierto del sistema (Figura 103) a una entrada se lleva a una de segundo orden (Ecuación 65).

$$G(s) = \frac{K * Wn^2}{s^2 + 2\varepsilon Wn + Wn^2} \quad (65)$$

Figura 103. Tercer orden Flexible Joint y respuesta ante una entrada.



$$V_{max} = 2.185 \quad K_{estabilizacion} = 1.863 \quad Ts(\text{Tiempo de estabilización}) = 0.8$$

$$M_p = 2.185 - 1.863 = 0.325 \quad \%MP = \frac{0.325}{1.863} = 0.17 \quad (66)$$

Con el %Mp (Ecuación 66) se halla el zita

$$\varepsilon = \sqrt[2]{\frac{\ln Mp^2}{\ln Mp^2 - \pi^2}} \quad \varepsilon = \sqrt[2]{\frac{\ln 0.17^2}{\ln 0.17^2 - \pi^2}} = 0.51 \quad (67)$$

Para hallar la frecuencia natural se usa:

$$Ts = \frac{4}{Wn * \varepsilon} \quad Wn = \frac{4}{Ts * \varepsilon} \quad Wn = \frac{4}{0.8 * 0.51} = 9.8 \quad (68)$$

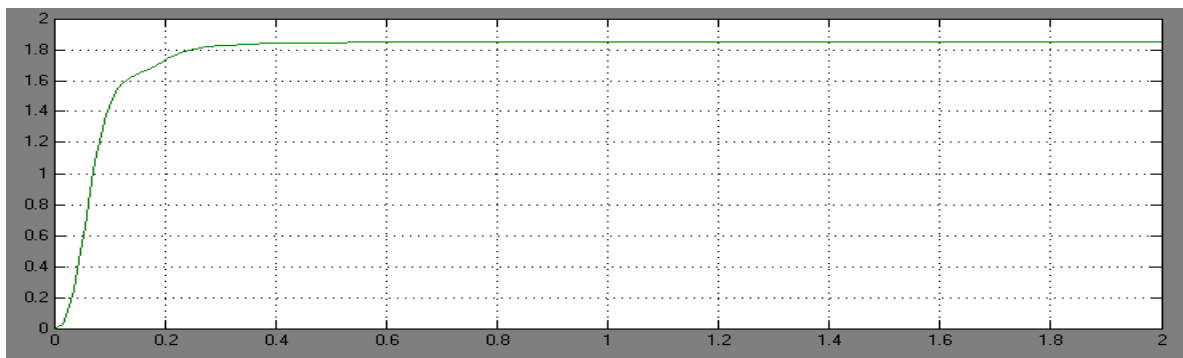
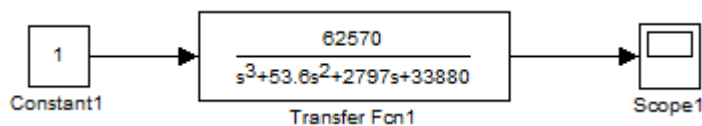
Al reemplazar todos los valores obtenidos en las ecuaciones 67 y 68 en la ecuación 65, se obtiene que la función de transferencia de segundo orden para el flexible joint es (Ecuación 17):

$$G(s) = \frac{178.92}{s^2 + 9.8s + 96.04} \quad (17)$$

Anexo B. Flexible Link segundo orden

Al ser de cuarto orden como el flexible joint, se realiza el mismo procedimiento se retira el integrador puro que tiene (Ecuación 12) y con la respuesta en lazo abierto (Figura 104) se lleva a segundo orden.

Figura 104. Tercer orden Flexible Link y respuesta ante una entrada.



$$V_{max} = 0 \quad K_{estabilizacion} = 1.847 \quad T_s(\text{Tiempo de estabilización}) = 0.5$$

El sistema tiene una respuesta sobreamortiguada, por tal motivo se determina un zeta mayor a 1

$$\varepsilon = 1.1 \quad (69)$$

Para hallar la frecuencia natural se usa:

$$T_s = \frac{4}{W_n * \varepsilon} \quad W_n = \frac{4}{T_s * \varepsilon} \quad W_n = \frac{4}{0.5 * 1.1} = 7.27 \quad (70)$$

Al reemplazar todos los valores de las ecuaciones 69 y 70 en la ecuación 65, se obtiene que la función de transferencia de segundo orden para el flexible link es (Ecuación 28):

$$G(s) = \frac{97.61}{s^2 + 16s + 52.85} \quad (28)$$

Anexo C. Código para generar cada uno de los conjuntos difusos Flexible Link y Flexible Joint

```
% controlador fuzzy en labview

x= -pi/4:pi/400:pi/4;

y= trapmf(x,[-pi/4 -pi/4 -pi/6 -pi/12]);
y1= gbellmf(x,[-pi/12 3.5 (-5*pi/24)]);

y2= trimf(x,[-pi/6 -pi/12 0]);
y3= gaussmf(x,[-pi/26 -pi/12]);

y4= trimf(x,[-pi/12 0 pi/12]);
y5= gaussmf(x,[-pi/26 0]);

y6= trimf(x,[0 pi/12 pi/6]);
y7= gaussmf(x,[-pi/26 pi/12]);

y8= trapmf(x,[pi/12 pi/6 pi/4 pi/4]);
y9= gbellmf(x,[pi/12 3.5 (5*pi/24)]);
```

```

plot(x,y,'g',x,y2,'g',x,y4,'g',x,y6,'g',x,y8,'g',x,y1,'b',x,y3,'b',x,y5,'
b',x,y7,'b',x,y9,'b')

x= -1:0.01:1;

y10= trapmf(x,[-1 -1 -0.66 -0.33]);
y11= gbellmf(x,[-0.33 3.5 -0.82]);

y12= trimf(x,[-0.66 -0.33 0]);
y13= gaussmf(x,[-0.16 -0.33]);

y14= trimf(x,[-0.33 0 0.33]);
y15= gaussmf(x,[0.16 0]);

y16= trimf(x,[0 0.33 0.66]);
y17= gaussmf(x,[0.16 0.33]);

y18= trapmf(x,[0.33 0.66 1 1]);
y19= gbellmf(x,[0.33 3.5 0.82]);

figure
plot(x,y10,'g',x,y12,'g',x,y14,'g',x,y16,'g',x,y18,'g',x,y11,'b',x,y13,'b
',x,y15,'b',x,y17,'b',x,y19,'b')

```

Anexo D. Código para generar cada uno de los conjuntos difusos Ball and Beam

```

% controlador fuzzy en labview

x= -0.45:0.01:0.45;

y= trapmf(x, [-0.45 -0.45 -0.3 -0.15]);
y1= gbellmf(x,[0.22 4.5 -0.45]);
%           ancho   caida centro
y2= trimf(x, [-0.3 -0.15 0]);
y3= gaussmf(x,[0.065 -0.15]);

y4= trimf(x, [-0.15 0 0.15]);
y5= gaussmf(x,[0.065 0]);

y6= trimf(x,[0 0.15 0.3]);
y7= gaussmf(x,[0.065 0.15]);

y8= trapmf(x, [0.15 0.3 0.45 0.45]);
y9= gbellmf(x,[0.22 4.5 0.45]);

plot(x,y,'g',x,y2,'g',x,y4,'g',x,y6,'g',x,y8,'g',x,y1,'b',x,y3,'b',x,y5,'
b',x,y7,'b',x,y9,'b')
TITLE('ERROR')

```

```

x= -1:0.01:1;

y10= trapmf(x,[-1 -1 -0.66 -0.33]);
y11= gbellmf(x,[-0.3 3 -0.82]);

y12= trimf(x,[-0.66 -0.33 0]);
y13= gaussmf(x,[-0.13 -0.33]);

y14= trimf(x,[-0.33 0 0.33]);
y15= gaussmf(x,[0.13 0]);

y16= trimf(x,[0 0.33 0.66]);
y17= gaussmf(x,[0.13 0.33]);

y18= trapmf(x,[0.33 0.66 1 1]);
y19= gbellmf(x,[0.3 3 0.82]);

figure
plot(x,y10,'g',x,y12,'g',x,y14,'g',x,y16,'g',x,y18,'g',x,y11,'b',x,y13,'b',
',x,y15,'b',x,y17,'b',x,y19,'b')
TITLE('DERIVADA DEL ERROR')

```